

A Discontinuous Galerkin Method for Gravity-Driven Viscous Fingering Instabilities in Porous Media.[☆]

G. Scovazzi^{a,*}, A. Gerstenberger^a, S. S. Collis^a

^a*Numerical Analysis and Applications Department, Sandia National Laboratories,
P.O. Box 5800, MS 1319, Albuquerque, NM 87185-1319, USA*

Abstract

We present a new approach to the simulation of gravity-driven viscous fingering instabilities in porous media flow. These instabilities play a very important role during carbon sequestration processes in brine aquifers. Our approach is based on a nonlinear implementation of the discontinuous Galerkin method, and possesses a number of key features. First, the method developed is inherently high order, and is therefore well suited to study unstable flow mechanisms. Secondly, it maintains high-order accuracy on completely unstructured meshes. The combination of these two features makes it a very appealing strategy in simulating the challenging flow patterns and very complex geometries of actual reservoirs and aquifers. This article includes an extensive set of verification studies on the stability and accuracy of the method, and also features a number of computations with unstructured grids and non-standard geometries.

Key words: Viscous fingering, discontinuous Galerkin method, gravity-driven flows, porous media flows.

1. Introduction

In order to reduce the carbon trace of industrialized countries, a number of strategies to remove the carbon dioxide from the atmosphere have been proposed. These approaches typically rely on pumping carbon dioxide deep in the Earth's crust, in regions where it can be stored indefinitely. A prevailing methodology for sequestering carbon dioxide is to inject it in brine aquifers extending deep in the subsurface, below layers of impermeable cap rock. The function of the cap rock is to prevent the carbon dioxide to resurface - in time - due to buoyancy forces. In a typical configuration, a region saturated with CO₂ is expected to form below the cap rock, and extend all the way to the water table of the aquifer. At the interface between the CO₂-saturated region and the aquifer, complex chemical dissolution processes take place, which can further help locking the carbon dioxide indefinitely into the aquifer. In addition, CO₂ can also mineralize into the porous medium, an additional mechanism of sequestration which is beyond the scope of this work.

In what follows, we present a new nonlinear implementation of the discontinuous Galerkin method, aimed at simulating the dissolution of CO₂ into the brine aquifer. In this process, finger-like plumes of brine very rich in carbon dioxide form and sink into the aquifer. The reason for the formation of such structures is that, as the CO₂ dissolves into the brine, it increases the brine's density. Consequently, a layer of denser brine forms on top of lighter brine, producing a clearly unstable mechanical configuration under the effect of gravity forces. The (linear) stability of gravity-driven viscous fingering has been investigated in the distant and recent past by [4, 6–8, 19, 33], which include a number of computations to support the theoretical findings. However, these computations almost exclusively rely on Fourier methods, or high-order finite-difference/finite-volume methods, which maintain high accuracy only on Cartesian, or nearly-Cartesian grids. These methods are perfectly suited for general studies on the dominant physics mechanisms involved in viscous fingering, but are not flexible enough to adapt to the very complex geometries of the subsurface. Contrary to a superficial intuition, the geometry of the subsurface could be quite complex in nature, and

[☆]Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin company, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

*Corresponding author

Email addresses: gscovaz@sandia.gov (G. Scovazzi)

Preprint submitted to Journal of Computational Physics

January 13, 2012

major computational engineering research groups in the field often face the same grid generation problems common to the majority of mechanical/aerospace engineering design firms.

More recently, an adaptive finite volume method has been presented in [16, 17] to attack gravity-driven flow instabilities. In this approach, structured Cartesian grids are used to perform computations, in conjunction with an adaptive multi-resolution mesh refinement strategy, driven by an error estimator. Although very promising, this methodology also falls under the category of structured methods, and if complex boundary or internal interfaces need to be simulated, then immersed/embedded boundary techniques or ghost-cell approaches will need to be used. Such choices may have the undesired effect of reducing accuracy near internal geometrical layers where sharp changes of rock permeability occur. The net effect would be then to reduce accuracy in regions that may dominate the dynamics of the entire system.

In the following work, we focus instead in using a discontinuous Galerkin discretization. We show that the discontinuous Galerkin approach allows for high-order computations on general unstructured grids, and therefore has the advantage to be a more flexible tool, especially when sharp changes in permeability occur in the subsurface. The proposed approach allows for an easy representation of discontinuity of material properties in the subsurface, and is an appealing alternative to structured finite-volume based techniques, whenever the geometrical complexity of the domain to be simulated is non-trivial.

The use of discontinuous Galerkin methods in subsurface flow has been explored in many recent works, particularly [5, 9, 14, 15, 18, 21–32]. However, these contributions have predominantly focused on the simulation of large-scale reservoirs, and furthermore they have been limited to low-order interpolations/approximations. The typical length-scale of large-scale reservoirs is kilometers, while in the case of carbon sequestration, finger plumes can be as small as few meters in cross section. In practice, the methodology we propose can also be applied in the near-well region of reservoir as a high-fidelity predictive tool. In addition, our work emphasizes the use of higher-order polynomials for high-fidelity solutions in practical engineering computations.

The article is organized as follows: Section 2 describes the governing equations of typical dissolution processes taking place during carbon dioxide sequestration in brine aquifers, the discontinuous Galerkin formulation is presented in Section 3, where the corresponding Euler-Lagrange equations are discussed, and stability/accuracy properties are outlined. Section 4 presents a extensive sequence of verification tests. Initially, simple decoupled linear problems are considered, followed by coupled nonlinear problems. Section 5 contains a number of viscous fingering computations that have been thoroughly documented in the literature, and are used as benchmarks for the implemented method. Finally, Section 6 offers a perspective on the true capabilities of the proposed method when complex geometry aquifers are the target application. Conclusions are summarized in Section 7.

2. Governing equations

Let Ω be an open set in \mathbb{R}^{n_d} (where n_d is the number of spatial dimensions). Ω is the computational domain with boundary $\Gamma = \partial\Omega$. The boundary Γ has outward-pointing normal \mathbf{n} . In what follows, vector fields will be denoted by boldface italic letters, and scalars fields by italic letters. For example, $\mathbf{x} \in \Omega$ is the position vector.

Dissolution and transport of CO_2 into a brine aquifer are governed by a classical advection-diffusion model, in which the equation of Darcy flow drives the fluid inside the porous medium. Isothermal conditions are assumed. More precisely, the mass conservation equation for species α in a two-species solution is given by

$$\partial_t(\phi \rho X_\alpha) + \nabla \cdot (\rho X_\alpha \mathbf{u}) = \nabla \cdot (\phi \tau \mathcal{D} \rho \nabla X_\alpha), \quad \alpha = 1, 2. \quad (1)$$

Here X_α is the concentration of species α with

$$X_1 + X_2 = 1, \quad (2)$$

and $\partial_t = \partial/\partial t$ denotes the Eulerian time derivative. In particular, X_1 is the concentration of carbon dioxide, ϕ denotes the porosity of the porous matrix in the subsurface, assumed independent of the pressure, $\tau \mathcal{D}$ is the brine effective diffusivity (the product of the *tortuosity* τ and *dispersion parameter* \mathcal{D} , both independent of X_α .) The density ρ of the brine solution is related to the brine composition according to

$$\rho = \left(\frac{X_1}{\rho_1} + \frac{X_2}{\rho_2} \right)^{-1} = \frac{\rho_1 \rho_2}{\rho_1 + (\rho_2 - \rho_1) X_1}, \quad (3)$$

where we have used (2) to express the dependence $\rho(X_1)$. The velocity \mathbf{u} with which the brine solution displaces in the porous medium is given by Darcy's law, namely,

$$\mathbf{u} = -\frac{\mathbf{K}}{\mu}(\nabla\varphi - \rho\mathbf{g}) . \quad (4)$$

The symbol \mathbf{K} indicates the permeability tensor, taken equal to a *constant* multiple of the identity, $\kappa\mathbf{I}$, for the sake of simplicity, μ is the viscosity of the solution, assumed to be constant (i.e., we assume that the concentration of CO_2 has little influence on the overall viscosity of the mixture). In addition, φ is the pressure, and \mathbf{g} is the gravity vector. Therefore the flow displaces driven by pressure gradients and gravitational forces.

Remark1. The dependency of ρ on the brine concentrations is fundamental in the development of viscous fingering instabilities in the problem considered. In fact, as the carbon dioxide dissolves into the brine, the overall density of the brine increases. Consequently, layers of denser brine form on top of lighter brine, creating an unstable mechanical configuration which spurs complex finger-like flow patterns.

It is possible to further reduce the equations (1)–(4) to a system of only two equations. In fact, using (2), the sum of equation (1) written for $\alpha = 1, 2$ yields

$$\partial_t(\phi\rho) + \nabla \cdot (\rho\mathbf{u}) = 0 . \quad (5)$$

Following the derivation in [7, 12, 17], we can now define a *porosity-scaled Lagrangian derivative*

$$D_t[\phi](\cdot) = \phi\partial_t(\cdot) + \mathbf{u} \cdot \nabla(\cdot) . \quad (6)$$

Subtracting the product of X_α and (5) from (1) yields

$$\rho D_t[\phi](X_\alpha) = \nabla \cdot (\rho\phi\tau\mathcal{D}\nabla X_\alpha) , \quad \alpha = 1, 2 , \quad (7)$$

Analogously, (5), can be written as

$$D_t[\phi](\rho) + \rho \nabla \cdot \mathbf{u} = 0 . \quad (8)$$

Applying $D_t[\phi]$ to the definition (3) of ρ , and using (7) yields

$$-\frac{D_t[\phi](\rho)}{\rho^2} = \sum_{\alpha=1}^2 \frac{D_t[\phi](X_\alpha)}{\rho_\alpha} = \sum_{\alpha=1}^2 \frac{\nabla \cdot (\rho\phi\tau\mathcal{D}\nabla X_\alpha)}{\rho\rho_\alpha} , \quad (9)$$

which can be substituted into (8) to obtain

$$\nabla \cdot \mathbf{u} = \sum_{\alpha=1}^2 \frac{\nabla \cdot (\rho\phi\tau\mathcal{D}\nabla X_\alpha)}{\rho_\alpha} , \quad (10)$$

or, using Darcy's law (4),

$$-\nabla \cdot \left(\frac{\kappa}{\mu} (\nabla\varphi - \rho\mathbf{g}) \right) = \sum_{\alpha=1}^2 \frac{\nabla \cdot (\rho\phi\tau\mathcal{D}\nabla X_\alpha)}{\rho_\alpha} . \quad (11)$$

Therefore, it is sufficient to solve the pressure equation (11) coupled with equation (1) written for X_1 , and use (2) whenever needed.

Remark2. The authors in [7, 11, 13, 17] point out that, due to the presence of the right hand side of (11), the proposed equations can produce instabilities if a perturbation in permeability is applied. In contrast, the standard Boussinesq approximation, for which the right hand side of (11) vanishes, can only produce instabilities if perturbative concentration gradients are introduced (as a result of porosity fluctuations, for example.)

2.1. Boundary and initial conditions

The boundary Γ can be further subdivided into portions on which different types of boundary conditions are enforced. In this case of the transport equation (1), inflow and outflow boundary conditions need to be imposed. At the inflow boundary, $\Gamma^{in} = \{\mathbf{x} \in \Gamma : \mathbf{u}(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) < 0\}$, the Dirichlet boundary condition on the concentration X_1 of carbon dioxide is enforced, that is

$$X_1|_{\Gamma^{in}} = \chi(\mathbf{x}, t) . \quad (12)$$

At the outflow boundary $\Gamma^{out} = \Gamma \setminus \Gamma^{in}$, the flow is allowed to exit the domain freely, and no value of the carbon dioxide concentration is imposed. Clearly, $\partial\Omega = \overline{\Gamma^{in} \cup \Gamma^{out}}$.

The pressure equation (11) is instead a standard Poisson equation of elliptic character driven by a forcing term dependent on concentration gradients: The typical boundary conditions are of Neumann and Dirichlet type. Assuming that the boundary $\partial\Omega$ is partitioned as $\partial\Omega = \overline{\Gamma_D \cup \Gamma_N}$, $\Gamma_D \cap \Gamma_N = \emptyset$, *pressure boundary conditions* are enforced on the *Dirichlet* boundary Γ_D , that is,

$$\phi|_{\Gamma_D} = \phi_D(\mathbf{x}, t) , \quad (13)$$

and *normal velocity boundary conditions* are enforced on the *Neumann* boundary Γ_N , namely

$$\mathbf{u} \cdot \mathbf{n}|_{\Gamma_N} = u_n(\mathbf{x}, t) . \quad (14)$$

Appropriate initial conditions on the concentration distribution

$$X_1(\mathbf{x}, t = 0) = X_1^0(\mathbf{x}) , \quad \text{on } \Omega , \quad (15)$$

complete the problem setting. We attack next the solution of the proposed equations using a high-order discontinuous Galerkin approach.

Remark3. Note that there is no need for a pressure initial condition, as the pressure plays the role of a Lagrange multiplier enforcing the flow balance condition (10), which, in the case of a single-species flow ($\nabla X_\alpha = 0$), collapses to a condition of flow incompressibility.

3. Discontinuous Galerkin formulation

The proposed discontinuous Galerkin (DG) discretization relies on an IMPES (IMplicit Pressures EXplicit Saturations) strategy for time integration, widely used in computational reservoir engineering. In our case, the saturation equations are replaced by the concentration equation (1). Hence, the concentration equation (1) is solved with a fourth-order explicit Runge-Kutta time integrator, and a global linear solve is applied to the pressure equation (11) at each stage of the Runge-Kutta method. Beside simplicity, the main reason for explicit time integration of the transport equation is that accurate time tracking is required in high-fidelity computations of complex transient instability phenomena. The alternative choice of using a fully coupled implicit method did not seem more advantageous, as time steps are in any case fairly small.

3.1. DG approximation spaces

Let us introduce a decomposition \mathcal{T}^e of the domain Ω into finite elements (of domain Ω_e and boundary Γ_e) such that each element does not overlap the others and $\overline{\Omega} = \bigcup_{e=1}^{n_{el}} \overline{\Omega_e}$ (where n_{el} is the total number of elements.) It is typical in DG formulations to introduce the following function spaces:

$$\mathcal{S} = \{v \in L^2(\Omega_e) : v|_{\Omega_e} \in H^1(\Omega_e), \forall \Omega_e \in \mathcal{T}^e\} , \quad (16)$$

$$\mathcal{V} = \{v \in (L^2(\Omega_e))^{n_d} : v|_{\Omega_e} \in (H^1(\Omega_e))^{n_d}, \forall \Omega_e \in \mathcal{T}^e\} . \quad (17)$$

Therefore, \mathcal{S} and \mathcal{V} are spaces of functions whose restrictions to each element belong to the Sobolev spaces $H^1(\Omega_e)$ and $(H^1(\Omega_1))^{n_d}$, respectively. \mathcal{S} and \mathcal{V} are often referred to as “broken spaces”, because of their discontinuous nature. Their discrete counterparts are

$$\mathcal{S}_h^p = \{v \in L^2(\Omega_e) : v|_{\Omega_e} \in \mathcal{P}^p(\Omega_e), \forall \Omega_e \in \mathcal{T}^e\} , \quad (18)$$

$$\mathcal{V}_h^p = \{\mathbf{v} \in (L^2(\Omega_e))^{n_d} : \mathbf{v}|_{\Omega_e} \in (\mathcal{P}^p(\Omega_e))^{n_d}, \forall \Omega_e \in \mathcal{T}^e\} , \quad (19)$$

where $\mathcal{P}^p(\Omega_e)$ (with $p \geq 1$) is the space of polynomials of at most order p , in the case of triangular/tetrahedral meshes, and the tensor product of polynomials of at most order p in the case of quadrilateral and hexahedral elements. In the context of the present work, the implementation of the DG method was performed using tensor products of Legendre polynomial spaces. Although the method can work equally well on grids of simplexes (triangles in two dimensions and tetrahedrons in three dimensions), the numerical computations presented here utilize only unstructured quadrilaterals.

3.2. DG notation

In what follows, ω typically denotes a portion of Ω , so that $\omega \subset \Omega$ (e.g., an element domain Ω_e .) The symbol γ indicates a portion of the boundary $\partial\omega$, which could also correspond with a portion of the boundary Γ . We denote by

$$(v, w)_\omega = \int_\omega v w \, d\omega , \quad (20)$$

$$(\mathbf{v}, \mathbf{w})_\omega = \int_\omega \mathbf{v} \cdot \mathbf{w} \, d\omega , \quad (21)$$

the $L^2(\omega)$ - and $(L^2(\omega))^{n_d}$ -inner products on the interior of ω and by

$$\langle v, w \rangle_\gamma = \int_\gamma v w \, d\gamma , \quad (22)$$

$$\langle \mathbf{v}, \mathbf{w} \rangle_\gamma = \int_\gamma \mathbf{v} \cdot \mathbf{w} \, d\gamma , \quad (23)$$

boundary functionals on γ . Now consider a generalized edge of the finite element mesh, and denote by $+$ and $-$ the element on either side of it. The DG approximation spaces are discontinuous across element interfaces, and it becomes useful to define the jump operator $[[\cdot]]$ and average operator $\{\cdot\}$ for *discontinuous* scalars and vector fields as follows:

$$[[v]] = v^+ \mathbf{n}^+ + v^- \mathbf{n}^- , \quad (24)$$

$$[[\mathbf{v}]] = \mathbf{v}^+ \cdot \mathbf{n}^+ + \mathbf{v}^- \cdot \mathbf{n}^- , \quad (25)$$

$$\{v\} = \frac{1}{2} (v^+ + v^-) , \quad (26)$$

$$\{\mathbf{v}\} = \frac{1}{2} (\mathbf{v}^+ + \mathbf{v}^-) . \quad (27)$$

These definitions are very valuable when deriving global DG formulations, and in particular, are invariant with respect of any intrinsic orientation of the element interfaces. However, in what follows, the superscript “+” will indicate the local element (the one over which interior integration is carried out), and the superscript “−” its neighbor: This notation will provide an easier understanding of the formulas. Note also that the following identity holds [1]:

$$[[\psi \mathbf{v}]] = \{\psi\} [[\mathbf{v}]] + [[\psi]] \cdot \{\mathbf{v}\} . \quad (28)$$

3.3. DG discretization of the transport equation (1)

Many different DG formulations can be adopted in the discretization of equations (1) and (11). In the case of equation (1) we opted for a pure upwind discretization [2, 20] of the convective term $\nabla(\rho \mathbf{u} X_1)$ and a local DG (LDG)

method [1, 3] for the diffusive term $\nabla \cdot (\phi \tau \mathcal{D} \rho \nabla X_1)$. Defining $v = \phi \tau \mathcal{D} \rho$, at each stage of the Runge-Kutta integrator, incremental equations of the type

$$\phi \delta(\rho X_1) + \nabla \cdot (\rho X_1 \mathbf{u} + \boldsymbol{\sigma}_d) = 0, \quad (29)$$

$$\boldsymbol{\sigma}_d = -v \nabla X_1, \quad (30)$$

need to be solved, with $\delta(z)$ representing an increment of the quantity z . The system (29)–(30) is in *mixed form*, and has a particularly convenient structure in the context of explicit time integration. Obviously, the initial conditions associated with (29) depend on the stage considered. Note that $U = \rho X_1$ is represented as a polynomial, adopting the so-called *conserved variable approach*. The value of the carbon dioxide concentration X_1 is then evaluated when needed at quadrature points using the definition of $\rho(X_1)$, namely,

$$X_1 = \frac{U}{U \left(1 - \frac{\rho_2}{\rho_1}\right) + \rho_2}. \quad (31)$$

Hence, the variational statement of the mixed form (29)–(30) reads:

Find $X_1 \in \mathcal{S}_h^p$ and $\boldsymbol{\sigma}_d \in \mathcal{V}_h^p$ such that, $\forall w \in \mathcal{S}_h^p$ and $\forall \boldsymbol{\tau} \in \mathcal{V}_h^p$,

$$0 = (w, \phi \delta(\rho X_1))_{\Omega_e} - (\nabla w, \rho \mathbf{u} X_1 + \boldsymbol{\sigma}_d)_{\Omega_e} + \langle w^+, \mathbf{n}_e^+ \cdot (\hat{\boldsymbol{\sigma}}_u + \hat{\boldsymbol{\sigma}}_d) \rangle_{\Gamma_e \setminus \Gamma} + \langle w^+, \mathbf{n}_e^+ \cdot (\hat{\boldsymbol{\sigma}}_u^\Gamma + \hat{\boldsymbol{\sigma}}_d^\Gamma) \rangle_{\Gamma_e \cap \Gamma}, \quad (32)$$

$$(\boldsymbol{\tau}, \boldsymbol{\sigma}_d)_{\Omega_e} = (\nabla \cdot (v \boldsymbol{\tau}), X_1)_{\Omega_e} - \langle v^+ \boldsymbol{\tau}^+ \cdot \mathbf{n}_e^+, \hat{X}_1 \rangle_{\Gamma_e \setminus \Gamma} - \langle v^+ \boldsymbol{\tau}^+ \cdot \mathbf{n}_e^+, \hat{X}_1^\Gamma \rangle_{\Gamma_e \cap \Gamma}. \quad (33)$$

The formulation is complete when the numerical fluxes are specified. Here, the upwind fluxes $\hat{\boldsymbol{\sigma}}_u$ and $\hat{\boldsymbol{\sigma}}_u^\Gamma$ at inter-element and domain boundaries are, respectively,

$$\hat{\boldsymbol{\sigma}}_u = \begin{cases} (\rho \mathbf{u} X_1)^-, & \text{if } \mathbf{n}_e^+ \cdot \{\mathbf{u}\} < 0, \\ (\rho \mathbf{u} X_1)^+, & \text{otherwise,} \end{cases} \quad (34)$$

and

$$\boldsymbol{\sigma}_u^\Gamma = \begin{cases} \rho(\chi) \mathbf{u} \chi, & \text{on } \Gamma^{in}, \\ (\rho \mathbf{u} X_1)^+, & \text{on } \Gamma \setminus \Gamma^{in}, \end{cases} \quad (35)$$

where χ is the Dirichlet boundary condition for X_1 . The LDG numerical fluxes are

$$\hat{\boldsymbol{\sigma}}_d = \{\boldsymbol{\sigma}_d\} + \varepsilon p^2 \left\{ \frac{v}{h_e} \right\} \llbracket X_1 \rrbracket, \quad (36)$$

$$\hat{X}_1 = \{X_1\}, \quad (37)$$

$$\hat{\boldsymbol{\sigma}}_d^\Gamma = \begin{cases} \boldsymbol{\sigma}_d^+ + \varepsilon p^2 \frac{v}{h_e} (X_1^+ - \chi), & \text{on } \Gamma^{in}, \\ \boldsymbol{\sigma}_d^+, & \text{on } \Gamma \setminus \Gamma^{in}, \end{cases} \quad (38)$$

$$\hat{X}_1^\Gamma = \begin{cases} \chi, & \text{on } \Gamma^{in}, \\ X_1^+, & \text{on } \Gamma \setminus \Gamma^{in}, \end{cases} \quad (39)$$

where h_e is a characteristic length scale, which can be defined as

$$h_e = \frac{\text{meas}(\Omega_e^+) + \text{meas}(\Omega_e^-)}{2 \text{meas}(\gamma_e)} \quad (40)$$

with γ_e the edge between the local element e^+ and the neighbor e^- , that is $\gamma_e = \overline{\Omega_e^+} \cap \overline{\Omega_e^-}$.

Remark4. The numerical diffusive flux $\hat{\sigma}_d$ is centered in nature, since it is computed by taking the average of the fluxes σ_d on either side of element interfaces.

Remark5 (penalty terms). The scalar ε is a penalty parameter, which was taken equal to 2.0 in all computations. Note also that the penalty term scales with the square of the polynomial order p .

Performing integration by parts, it is possible to derive the Euler-Lagrange equations relative to the specific element e , which elucidate the numerical approach:

$$\begin{aligned} 0 &= (w, \phi \delta(\rho X_1) + \nabla \cdot (\rho X_1 \mathbf{u} + \sigma_d))_{\Omega_e} \\ &\quad + \langle w^+, ((\rho \mathbf{u} X_1)^- - (\rho \mathbf{u} X_1)^+) \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e^{in} \setminus \Gamma} + \langle w^+, (\rho(\chi) \chi - \rho^+ X_1^+) \mathbf{u}^+ \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e^{in} \cap \Gamma} \\ &\quad - \langle w^+, (\sigma_d^+ - \hat{\sigma}_d) \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \setminus \Gamma} - \langle w^+, (\sigma_d^+ - \hat{\sigma}_d^\Gamma) \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \cap \Gamma}, \\ 0 &= (\tau, \sigma_d - \nu \nabla X_1)_{\Omega_e} + \langle \nu^+ \tau^+ \cdot \mathbf{n}_e^+, X_1^+ - \hat{X}_1 \rangle_{\Gamma_e \setminus \Gamma} + \langle \nu^+ \tau^+ \cdot \mathbf{n}_e^+, X_1^+ - \hat{X}_1 \rangle_{\Gamma_e \cap \Gamma}, \end{aligned} \quad (41)$$

where $\Gamma_e^{in} = \{\mathbf{x} \in \Gamma_e : \mathbf{u}(\mathbf{x}) \cdot \mathbf{n}_e^+(\mathbf{x}) < 0\}$ is the element inflow boundary. Therefore, the DG formulation is weakly enforcing (29)–(30) on element interiors, and, *separately*, the continuity with respect to the *upwind* advective flux and the *centered* diffusive flux at element interfaces. This formulation is stable in the advective–diffusive and advective–dominated case, and its stability and convergence properties are studied in the linear case with full detail in [1–3] and references therein.

Remark6 (local and global conservation). It is not very difficult to realize that the derived formulation is locally conservative (taking w equal to unity over Ω_e in (29)–(30).) Also note that, summing all element contributions, the boundary terms of each element telescope and yield a global conservation statement.

3.4. DG discretization of the pressure equation (11)

In the case of the elliptic pressure equation (11), a symmetric interior penalty (SIPG) DG formulation was adopted [1, 10]. An alternative could have been an LDG formulation, which would have however required the implementation of discrete lifting operators which are more complicated to implement. Other authors [21, 22, 24, 25, 31] have also proposed non-symmetric interior penalty (NIPG) formulations for the same elliptic operators. The symmetric interior penalty formulation proved efficient in its implementation and robust in computations. In mixed form, the SIPG formulation varies only slightly from an LDG approach.

3.4.1. Mixed local formulation

Taking $\varphi = \sum_{\alpha=1}^2 1/\rho_\alpha \nabla \cdot (\rho \phi \tau \mathcal{D} \nabla X_1)$ the mixed form of the pressure equation reads

$$\nabla \cdot \mathbf{u} = \varphi, \quad (42)$$

$$\mathbf{u} = -\frac{\kappa}{\mu} (\nabla \varphi - \rho \mathbf{g}). \quad (43)$$

Indicating with $\langle w, \varphi \rangle_{\Omega_e}$ the linear form (in w) obtained from the LDG discretization of φ described in Section 3.3, the SIPG formulation of (11) is:

Find $\varphi \in \mathcal{S}_h^p$ and $\mathbf{u} \in \mathcal{V}_h^p$ such that, $\forall w \in \mathcal{S}_h^p$ and $\forall \tau \in \mathcal{V}_h^p$,

$$\langle w, \varphi \rangle_{\Omega_e} = -(\nabla w, \mathbf{u})_{\Omega_e} + \langle w^+, \hat{\mathbf{u}} \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \setminus \Gamma} + \langle w^+, \hat{\mathbf{u}}^\Gamma \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \cap \Gamma}, \quad (44)$$

$$(\tau, \mathbf{u})_{\Omega_e} = \left(\nabla \cdot \left(\frac{\kappa}{\mu} \tau \right), \varphi \right)_{\Omega_e} + \left(\tau, \frac{\kappa}{\mu} \rho \mathbf{g} \right)_{\Omega_e} - \left\langle \frac{\kappa^+}{\mu^+} \tau^+ \cdot \mathbf{n}_e^+, \hat{\varphi} \right\rangle_{\Gamma_e \setminus \Gamma} - \left\langle \frac{\kappa^+}{\mu^+} \tau^+ \cdot \mathbf{n}_e^+, \hat{\varphi}^\Gamma \right\rangle_{\Gamma_e \cap \Gamma}, \quad (45)$$

with the SIPG numerical fluxes defined as

$$\hat{\mathbf{u}} = \left\{ -\frac{\kappa}{\mu} (\nabla \varphi - \rho \mathbf{g}) \right\} + \varepsilon p^2 \left\{ \frac{\kappa}{\mu h_e} \right\} \llbracket \varphi \rrbracket, \quad (46)$$

$$\hat{\varphi} = \{\varphi\}, \quad (47)$$

$$\hat{\mathbf{u}}^\Gamma = \begin{cases} -\frac{\kappa}{\mu} (\nabla \varphi - \rho \mathbf{g}) + \varepsilon p^2 \frac{\kappa}{\mu h_e} (\varphi - \varphi_D) \mathbf{n}, & \text{on } \Gamma_D, \\ u_n \mathbf{n}, & \text{on } \Gamma_N, \end{cases} \quad (48)$$

$$\hat{\varphi}^\Gamma = \begin{cases} \varphi_D, & \text{on } \Gamma_D, \\ \varphi, & \text{on } \Gamma_N. \end{cases} \quad (49)$$

Again, the Euler Lagrange equations associated with the SIPG formulation can be obtained using integration by parts, and read

$$\begin{cases} 0 = (w, \nabla \cdot \mathbf{u})_{\Omega_e} - \langle w, \varphi \rangle_{\Omega_e} + \langle w^+, (\hat{\mathbf{u}} - \mathbf{u}^+) \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \setminus \Gamma} + \langle w^+, (\hat{\mathbf{u}}^\Gamma - \mathbf{u}^+) \cdot \mathbf{n}_e^+ \rangle_{\Gamma_e \cap \Gamma}, \\ 0 = (\boldsymbol{\tau}, \mathbf{u} + \kappa/\mu (\nabla \varphi - \rho \mathbf{g}))_{\Omega_e} + \langle \kappa^+/\mu^+ \boldsymbol{\tau}^+ \cdot \mathbf{n}_e^+, \hat{\varphi} - \varphi^+ \rangle_{\Gamma_e \setminus \Gamma} + \langle \kappa^+/\mu^+ \boldsymbol{\tau}^+ \cdot \mathbf{n}_e^+, \hat{\varphi}^\Gamma - \varphi^+ \rangle_{\Gamma_e \cap \Gamma}, \end{cases} \quad (50)$$

Remark7 (local and global conservation). Also the SIPG formulation can be proved to be locally and globally conservative [1].

3.4.2. Primal global formulation

A global *primal* formulation is derived next from the previous mixed statement of the SIPG approach. The global primal formulation has the advantage of involving only one unknown field (namely, the pressure), and was implemented for numerical computations. Derivations of the type that follows can also be found in the available literature (see [1] and references therein.) There, however, not much emphasis is placed on the treatment of the gravity terms. Because gravity plays a crucial role in the context of the proposed work, algebraic manipulations are presented in full detail. Substituting $\boldsymbol{\tau} = \nabla w$ in the second of (50), and omitting the + superscripts, as there is no risk of confusion, yields

$$0 = (\nabla w, \mathbf{u} + \kappa/\mu (\nabla \varphi - \rho \mathbf{g}))_{\Omega_e} + \langle \kappa/\mu \nabla w \cdot \mathbf{n}_e, \hat{\varphi} - \varphi \rangle_{\Gamma_e \setminus \Gamma} + \langle \kappa/\mu \nabla w \cdot \mathbf{n}_e, \hat{\varphi}^\Gamma - \varphi \rangle_{\Gamma_e \cap \Gamma} \quad (51)$$

Denoting by \mathcal{E} the set of all mesh edges (in two dimensions, or faces in three dimensions), summing over all the elements, applying the flux definitions (48)–(49), using the identity (28), and observing that, by definition, $\{\hat{\varphi} - \varphi\} = 0$ and $\llbracket \hat{\varphi} \rrbracket = 0$, yields

$$\begin{aligned} 0 &= (\nabla w, \mathbf{u} + \kappa/\mu (\nabla \varphi - \rho \mathbf{g}))_{\Omega} + \langle 1, \llbracket \kappa/\mu \nabla w (\hat{\varphi} - \varphi) \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} + \langle \kappa/\mu \nabla w \cdot \mathbf{n}, \varphi_D - \varphi \rangle_{\Gamma_D} \\ &= (\nabla w, \mathbf{u} + \kappa/\mu (\nabla \varphi - \rho \mathbf{g}))_{\Omega} + \langle \llbracket \kappa/\mu \nabla w \rrbracket, \{\hat{\varphi} - \varphi\} \rangle_{\mathcal{E} \setminus \Gamma} + \langle \{\kappa/\mu \nabla w\}, \llbracket \hat{\varphi} - \varphi \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} + \langle \kappa/\mu \nabla w \cdot \mathbf{n}, \varphi_D - \varphi \rangle_{\Gamma_D} \\ &= (\nabla w, \mathbf{u})_{\Omega} + (\nabla w, \kappa/\mu (\nabla \varphi - \rho \mathbf{g}))_{\Omega} - \langle \{\kappa/\mu \nabla w\}, \llbracket \varphi \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} - \langle \kappa/\mu \nabla w \cdot \mathbf{n}, \varphi - \varphi_D \rangle_{\Gamma_D} \end{aligned} \quad (52)$$

Consider now the sum of (44) over all the elements of the computational domain

$$\begin{aligned} (\nabla w, \mathbf{u})_{\Omega} &= -\langle w, \varphi \rangle_{\Omega} \\ &\quad - \langle \llbracket w \rrbracket, \{\kappa/\mu (\nabla \varphi - \rho \mathbf{g})\} \rangle_{\mathcal{E} \setminus \Gamma} + \langle \llbracket w \rrbracket, \varepsilon p^2 \{\kappa/(\mu h_e)\} \llbracket \varphi \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} \\ &\quad - \langle w, \kappa/\mu (\nabla \varphi - \rho \mathbf{g}) \cdot \mathbf{n} \rangle_{\Gamma_D} + \langle w, \varepsilon p^2 \kappa/(\mu h_e) (\varphi - \varphi_D) \rangle_{\Gamma_D} \\ &\quad + \langle w, u_n \rangle_{\Gamma_N}. \end{aligned} \quad (53)$$

Substituting (52) into (53), the global SIPG formulation is obtained:

Find $\varphi \in S_h^p$ such that, $\forall w \in S_h^p$,

$$B(w, \varphi) = L(w), \quad (54)$$

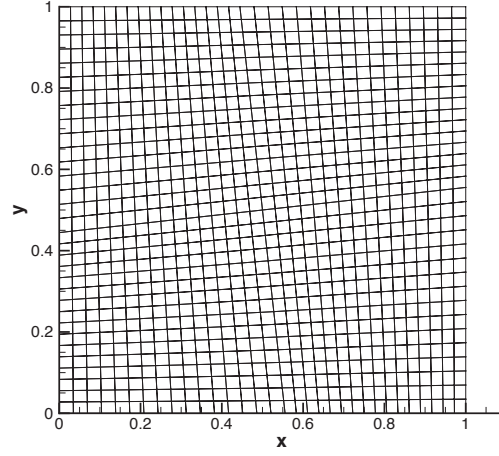


Figure 1: Sketch of the 32×32 skewed mesh (level 4), used to perform the convergence tests in Figure 2.

with

$$\begin{aligned}
 B(w, \varphi) = & (\nabla w, \kappa/\mu \nabla \varphi)_{\Omega} \\
 & - \langle \{\kappa/\mu \nabla w\}, \llbracket \varphi \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} - \langle \llbracket w \rrbracket, \{\kappa/\mu \nabla \varphi\} \rangle_{\mathcal{E} \setminus \Gamma} + \langle \llbracket w \rrbracket, \varepsilon p^2 \{\kappa/(\mu h_e)\} \llbracket \varphi \rrbracket \rangle_{\mathcal{E} \setminus \Gamma} \\
 & - \langle \kappa/\mu \nabla w \cdot \mathbf{n}, \varphi \rangle_{\Gamma_D} - \langle w, \kappa/\mu \nabla \varphi \cdot \mathbf{n} \rangle_{\Gamma_D} + \langle w, \varepsilon p^2 \kappa/(\mu h_e) \varphi \rangle_{\Gamma_D}, \tag{55}
 \end{aligned}$$

and

$$\begin{aligned}
 L(w) = & (\nabla w, (\kappa/\mu) \rho \mathbf{g})_{\Omega} + \langle w, \varphi \rangle_{\Omega} \\
 & - \langle \llbracket w \rrbracket, \{(\kappa/\mu) \rho \mathbf{g}\} \rangle_{\mathcal{E} \setminus \Gamma} \\
 & - \langle w, \kappa/\mu \rho \mathbf{g} \cdot \mathbf{n} \rangle_{\Gamma_D} - \langle \kappa/\mu \nabla w \cdot \mathbf{n}, \varphi_D \rangle_{\Gamma_D} + \langle w, \varepsilon p^2 \kappa/(\mu h_e) \varphi_D \rangle_{\Gamma_D} \\
 & - \langle w, u_n \rangle_{\Gamma_N}. \tag{56}
 \end{aligned}$$

Remark8. The bilinear form $B(\cdot, \cdot)$ is clearly symmetric, and, as such, adjoint consistent [1]. Adjoint consistency ensures the optimal convergence rates of the discretization.

4. Preliminary verification studies

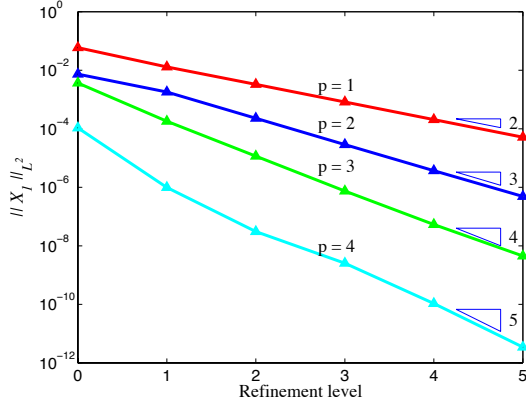
As a first step in the evaluation of the performance of the proposed method, standard convergence tests are performed. First, the case of an uncoupled advection-diffusion problem driven by an assigned velocity \mathbf{u} is considered. Then, the case of an uncoupled Poisson problem for the SIPG formulation of the pressure equation is evaluated, and finally a convergence test for the fully coupled equations is considered. All tests are two-dimensional.

4.1. Uncoupled transport test cases

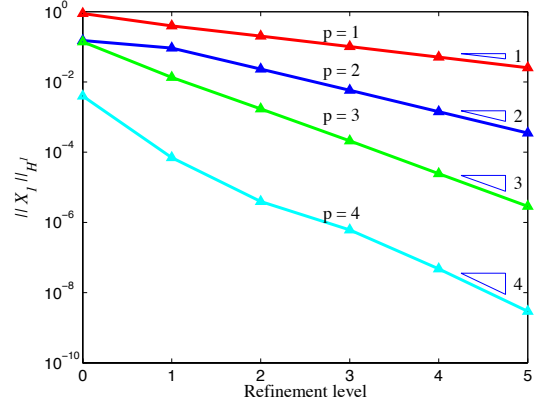
Two tests of pure transport are presented next: For both, the velocity field \mathbf{u} is assigned as *datum* rather than being computed from the pressure solve. The first test is a pure diffusion transient on a non-cartesian mesh. Equation (1) is solved with $\mathbf{u} = \mathbf{0}$, $\rho_1 = \rho_2 = \rho = 1$, $\phi = 1$, $\tau \mathcal{D} = 0.01$, over the unit square domain $\Omega = [0, 1] \times [0, 1]$. The initial solution is given as

$$X_1(x, y, t = 0) = 0.5 + 0.2 \sin(2\pi x) \sin(2\pi y), \tag{57}$$

and Dirichlet boundary conditions $X_1 = 0.5$ are applied over the entire boundary Γ of Ω . The domain is meshed with

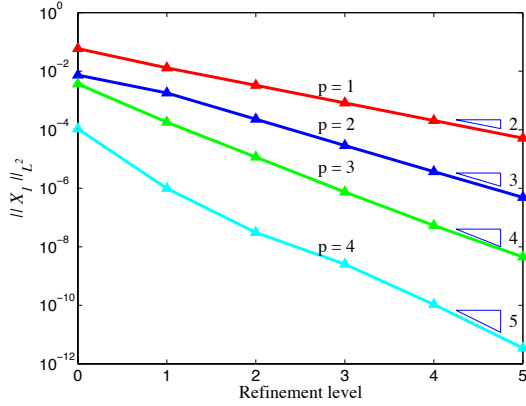


(a) L^2 -norm of the error.

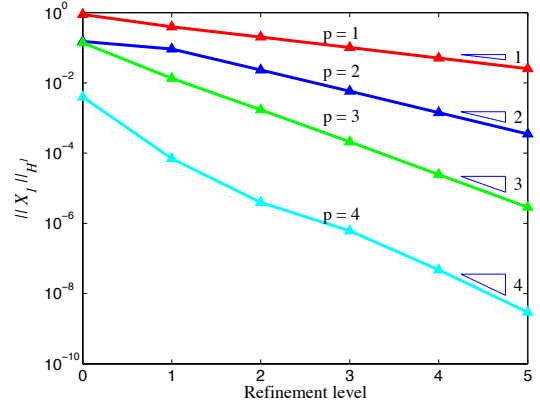


(b) H^1 -seminorm of the error.

Figure 2: Pure diffusion test: Convergence results in terms of the L^2 -norm and H^1 -seminorm of the error, respectively. Slopes of optimal convergence rates are indicated with triangles.



(a) L^2 -norm of the error.



(b) H^1 -seminorm of the error.

Figure 3: Advection-diffusion test: Convergence results in terms of the L^2 -norm and H^1 -seminorm of the error, respectively. Slopes of optimal convergence rates are indicated with triangles.

a computational grid which is slightly stretched and twisted to prevent error cancellations and super-convergence. The initial mesh is given by 2×2 elements, and four levels of grid refinement are considered, up to 32×32 elements. The mesh at the 32×32 -refinement level is depicted in Figure 1. The analytic solution of this problem is

$$X_1(x, y, t) = 0.5 + 0.2 e^{-8\pi^2 \tau \mathcal{D} t} \sin(2\pi x) \sin(2\pi y). \quad (58)$$

The solution is advanced in time for 100 time steps of size 10^{-5} . Such small time steps ensure that the time integration error is negligible with respect to the spatial error. Results for the L^2 -norm and H^1 -seminorm of the error are plotted in Figure 2. The convergence rates are optimal as predicted by the theory [1, 3], that is, the solution converges at a rate $p + 1$ in the L^2 -norm and p in the H^1 -seminorm.

The second test is again a pure transport case, but this time with both advection and diffusion. The problem is solved on the unit square, with periodic boundary conditions, and with the same physical parameters as in the previous test, with the exception of the velocity $\mathbf{u} = (1.0, 0.5)$. The initial mesh is a 2×2 -element Cartesian grid, and four levels of grid refinement are applied. The initial condition is given again by (57), and the exact solution is now

$$X_1(x, y, t) = 0.5 + 0.2 e^{-8\pi^2 \tau \mathcal{D} t} \sin(2\pi(x - 1.0)) \sin(2\pi(y - 0.5)). \quad (59)$$

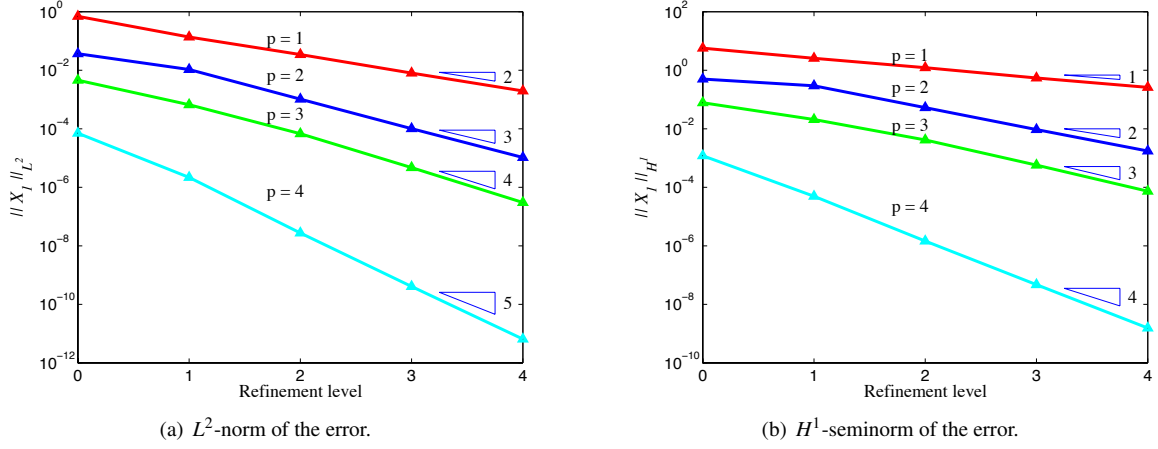


Figure 4: Uncoupled Darcy flow test: Convergence results in terms of the L^2 -norm and H^1 -seminorm of the error, respectively. Slopes of optimal convergence rates are indicated with triangles.

The solution in time for 1000 time steps of size 10^{-6} , to ensure again that the time discretization error is negligible with respect to the spatial discretization error. Figure 3 shows the convergence rates in the L^2 -norm and H^1 -seminorm. As predicted by the theory, the method converges optimally.

In summary, the results on this section show that the implemented LDG formulation for the transport equation (1) is optimally convergent in the diffusive and advective-diffusive case.

4.2. Uncoupled Darcy test case

An uncoupled test was performed also for the Darcy pressure equation. In this case, equation (11) was solved on the unit square domain, with $\mathbf{g} = \mathbf{0}$ and the right hand side replaced by the forcing

$$\varphi(x, y) = -8\pi^2 \sin(2\pi x) \sin(2\pi y). \quad (60)$$

The following exact solution holds in the case $\kappa = \mu = 1.0$ and homogenous Dirichlet boundary conditions are applied:

$$\varphi(x, y) = \sin(2\pi x) \sin(2\pi y). \quad (61)$$

The base mesh is composed of 2×2 elements, and four levels of grid refinement are used, the last of which is depicted in Figure 1. Again, the optimal convergence rates for the uncoupled SIPG formulation [1, 10] can be appreciated in Figure 4.

4.3. Coupled test cases

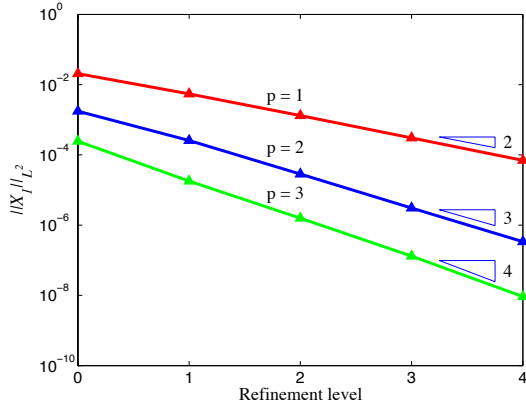
We conclude this section with a coupled two-dimensional test case, using the manufactured solution approach. This test is important in quantifying the quality of the overall numerical method. Because the system of coupled equations is nonlinear, the theory of DG formulations is helpful but not conclusive. The test case parameters are $\rho_1 = 1.0$, $\rho_2 = 10.0$, $\kappa = \mu = 1.0$, $\phi = 0.5$, $\mathbf{g} = \mathbf{0}$, $\tau\mathcal{D} = 0.001$. Appropriate right hand sides are added to the discretization of (1) and (11), so that the following exact solution is obtained:

$$X_1(x, y, t) = 0.75 + 0.25 \sin(2\pi(x - 1.0 t / \phi)), \quad (62)$$

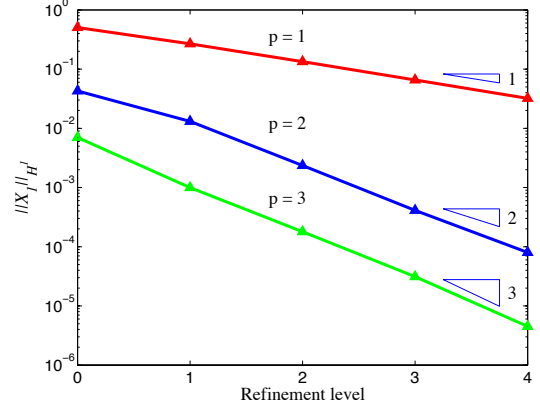
$$\varphi(x, y, t) = 2 - x, \quad (63)$$

$$\mathbf{u}(x, y, t) = \mathbf{e}_x, \quad (64)$$

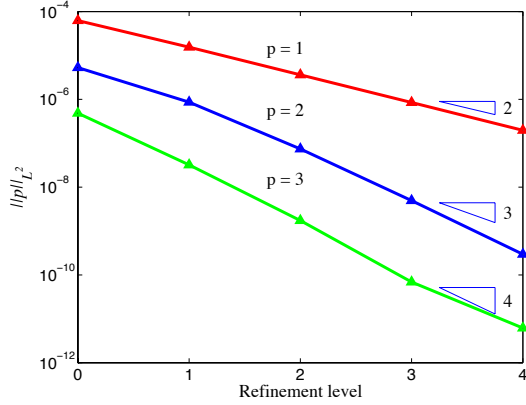
where \mathbf{e}_x is the unit vector along x . The base mesh has 4×4 elements, and four levels of refinement are applied. The plots in Figure 5 show that also in the nonlinear case the method converges optimally. Note in particular that although the pressure solution is a stationary solution, the right hand side of the pressure equation contains terms



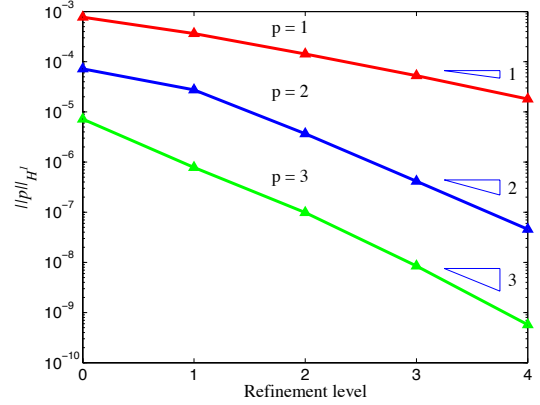
(a) L^2 -norm of the concentration error.



(b) H^1 -seminorm of the concentration error.



(c) L^2 -norm of the pressure error.



(d) H^1 -seminorm of the pressure error.

Figure 5: Two-dimensional coupled convergence test: Convergence results in terms of the L^2 -norm and H^1 -seminorm of the error for pressure and carbon dioxide concentration. Slopes of optimal convergence rates are also indicated. Note that the reduction of the slope of the L^2 -norm of the error for the case $p = 3$ and the highest level of refinement is due to the proximity with the roundoff threshold.

that are unsteady and, due to approximation errors, do not cancel perfectly. For this reason, an error is found in the pressure solution. Observe also the reduced slope of the convergence rate of the pressure L^2 -error for $p = 3$ between the refinement levels 3 and 4: This is due to the proximity of the roundoff threshold: The pressure is indeed converging optimally for coarser levels of refinement.

5. Advanced refinement and verification studies on viscous fingering instabilities

Results from a mesh-size/polynomial-order refinement study on a viscous fingering instability problem are presented next. The problem setup in [17] for viscous fingering instabilities is used as a benchmark for the proposed DG approach. The specified fluid properties correspond to 45 °C of temperature and 100 bar of pressure, as encountered in the Earth crust near 1000 m of depth.

The domain is $\Omega = [0, 1] \times [-4, 0]$ (dimensions in m^2), with the following physical parameters: $\mu = 0.5947 \times 10^{-3} \text{ Pa}$, $\tau\mathcal{D} = 2 \times 10^{-9} \text{ m}^2\text{s}^{-1}$, $\phi = 0.3$, $\mathbf{g} = [0.0, -g]$, $g = 9.81 \text{ ms}^{-2}$, and average permeability $\bar{k} = 10 \text{ D} = 9.869233 \times 10^{-12} \text{ m}^2$. The actual distribution of permeability was obtained by adding a 0.25% perturbation (in the

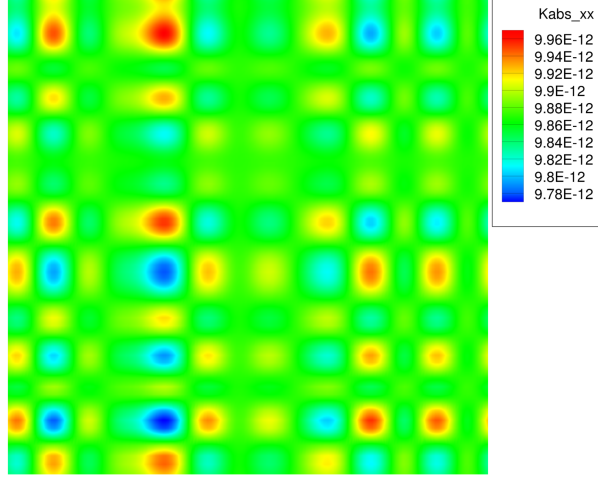


Figure 6: Randomized permeability field used to spark viscous fingering instabilities. The domain size is the unit square (in meters.)

L^2 -norm) to the average permeability. Namely,

$$\kappa = \bar{\kappa} \left(1 + 0.025 \frac{2}{N} \sum_{n=1}^N \sin \left(2\pi n \left(\frac{x}{L_x} - rnd_x \right) \right) \sin \left(2\pi n \left(\frac{y}{L_y} - rnd_y \right) \right) \right) \quad (65)$$

with $N = 8$, $L_x = 1.0$ m, $L_y = 4.0$ m, and rnd_x and rnd_y two uniformly distributed random numbers in the interval $[0, 1]$. In the limit for $N \rightarrow \infty$, the permeability κ would be perturbed with white noise of intensity 0.025. The sum was truncated after the first eight terms so that it can be fully resolved within the DG approximation spaces used, at least near the top layer. The resulting permeability field is depicted in Figure 6.

Remark9. The size of the computational domain is chosen so that it contains a sufficiently large multiple of the fastest growing wavelength of the instability (in the horizontal direction.) Linear theory analysis [17, 33] predicts that the fastest growing mode corresponds to the wavelength

$$\lambda_c = 96.23 \frac{\phi \mu \tau \mathcal{D}}{\kappa \Delta \rho g} \approx 0.03 \text{ m} , \quad (66)$$

which yields that about 30 wavelengths of the most unstable (i.e., fastest growing, or *critical*) mode fit along the horizontal dimension.

Remark10. The proposed definition of the permeability perturbation allows for a very tight control of the distribution and intensity under mesh and polynomial order refinement. There are other ways to perturb the permeability: For example, the authors in [17] used a random distribution of constants values in each cell of the finite volume grid. Because a perturbation distribution of this type changes as the mesh is refined, the authors in [17] had to rescale the intensity of the perturbation, to maintain approximately constant the instability onset time. Consequently, the problem statement in [17] changes slightly as the mesh is refined. In our case, the permeability perturbation is analytic and kept fixed as the mesh is refined or the polynomial order is increased, so that only the approximation error on the perturbation is the difference between various realizations of the same test.

The saturated carbon dioxide mass fraction is $\chi = 0.049309$, the density increase due to fully saturated carbon dioxide $\Delta \rho = 10.45 \text{ kg m}^{-3}$, which corresponds, substituting χ and $\rho_2 = 994.56 \text{ kg m}^{-3}$ into (3), to $\rho_1 = 1260.35 \text{ kg m}^{-3}$. The boundary conditions for the pressure equation are given as *no flow* ($\mathbf{u} \cdot \mathbf{n} = 0$) on all sides of the rectangular domain except the bottom boundary, where the pressure is set to the reference value 0 (the pressure is in general defined up to a constant). The boundary conditions for the transport equation are specified as homogeneous Neumann conditions on all but the top side, where the carbon dioxide concentration is in saturated conditions, that is $X_1 = \chi = 0.049309$. The initial distribution of carbon dioxide concentration is zero over the entire domain.

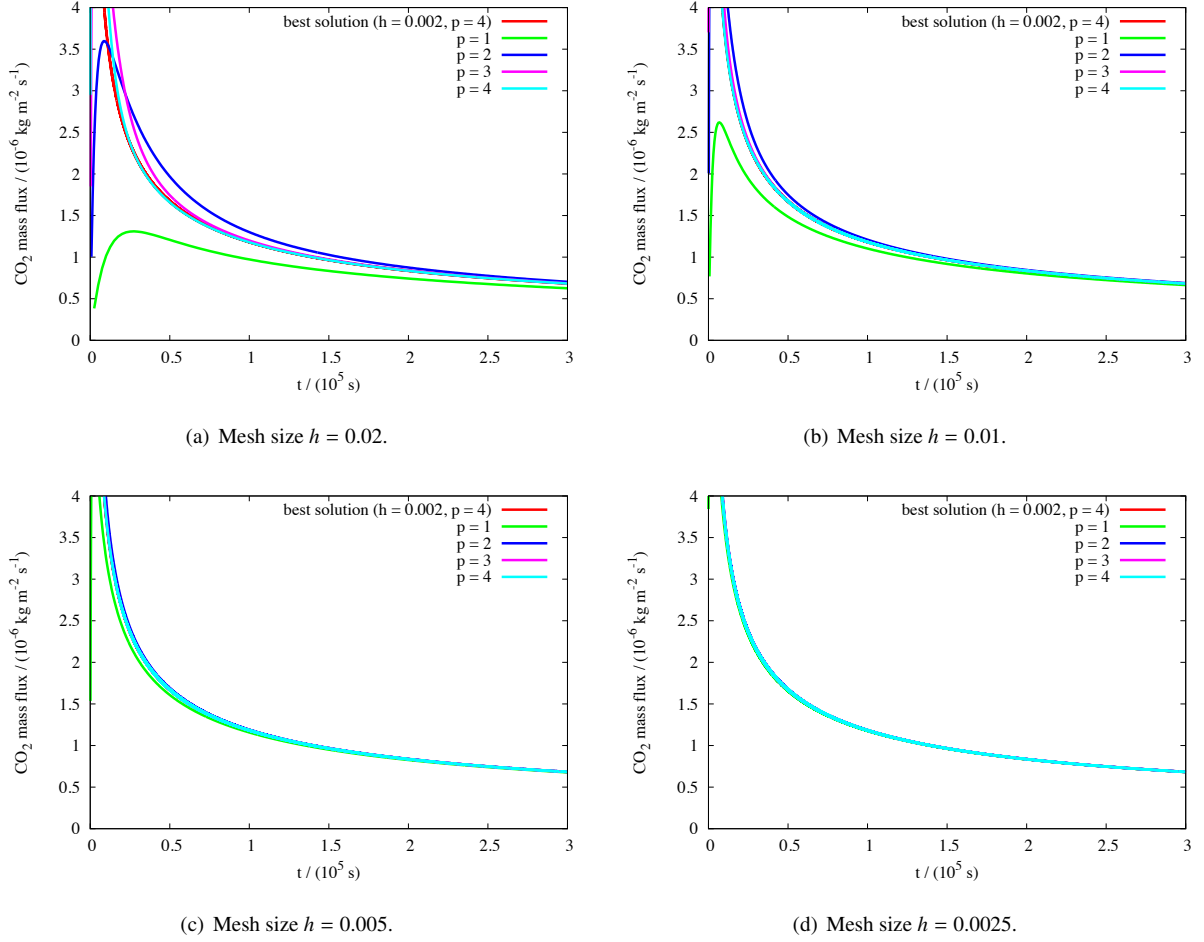


Figure 7: Time histories of the carbon dioxide flux at the top of the computational domain. One-dimensional study on the effect of resolution.

5.1. A convergence study on the onset time of viscous instabilities

Viscous fingering instabilities are bound to occur in the problem setup considered since as the carbon dioxide dissolves into the brine (water), the density of the water increases. As more and more carbon dioxide dissolves, a layer of heavy brine forms on top of a layer of lighter brine, a configuration which is unstable and spurs the formation of finger-like plumes sinking toward the bottom of the domain, as also shown in Figure 13.

The initial system dynamics is in reality very simple: The flow is at rest and any change in carbon dioxide concentration occurs only due to diffusion processes. At the initial time, the pressure balances the gravity exactly, that is the right hand side of (11) vanishes, and the system is in an (unstable) equilibrium configuration. As time progresses and carbon dioxide dissolves and diffuses into the reservoir, the right hand side of (11) becomes different from zero, and the specified perturbation in permeability causes minute flow velocities to form into the reservoir. These advection mechanisms are sufficient to start the formation of viscous fingers.

The very early stages of the dynamics is controlled by linear mechanisms, and only later on the system becomes fully nonlinear and chaotic behavior emerge. Because very small approximations errors in the permeability distribution can lead, after some time, to completely different solutions, discussing about numerical convergence can only be done in terms of spatial- or time-averaged quantities of interest. The flux of carbon dioxide entering the reservoir from the top boundary is a good engineering measure of the performance of the reservoir at sequestering and storing CO_2 .

Because linear dynamics phenomena prevail in the system until the onset of instability, it is actually possible to undertake a convergence study. Generally speaking, the onset time of viscous fingering is the time at which the

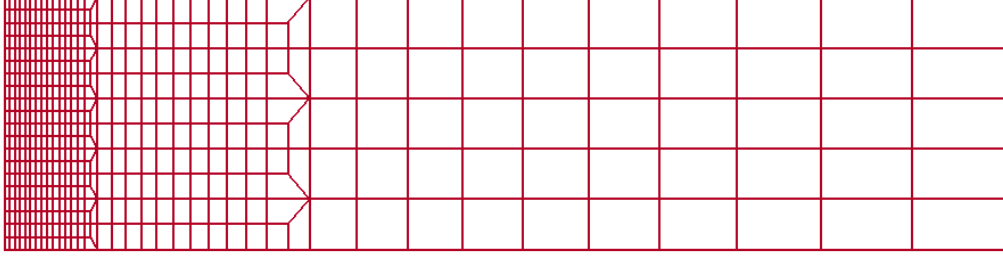


Figure 8: Sketch of the base mesh (X_0) used for the computation of onset times, rotated counterclockwise by 90 degrees.

systems diverts from a purely diffusive dynamics, and advection mechanisms start to dominate. For example, it can be defined as the time at which the actual flux of carbon dioxide is larger by more than 1% from the flux in absence of advective mechanisms.

5.1.1. Resolution requirements for the initial diffusion layer

The aim of the the next test cases is to quantify what is a sufficient level of resolution to accurately capture the initial diffusion mechanisms. The concentration layer that initially forms near the top boundary has all the characteristics of a “thermal shock” layer in heat transfer computations, and requires fairly refined meshes. First though, observe that, since ρ depends on X_1 according to (3), even if the velocity \mathbf{u} were to vanish, equation (1) would still be nonlinear.

In the diffusive layer, at least initially, the flow is nearly one dimensional. Therefore, to quantify the resolution needed to capture the diffusive layer, one-dimensional computations are sufficient. A number of one-dimensional tests (along the vertical direction of the domain) are shown in Figures 11 and 7, for uniform meshes of different resolution. In particular, Figures 11 and 7 contain the same information, presented in two different ways. Figure 11 shows for each polynomial order the effect of mesh refinement. Figure 7 shows the effect of incrementing the polynomial order at fixed mesh resolution. It can be easily appreciated that the time history of the flux converges as we increase the resolution, either changing the grid spacing or the polynomial order used. Summarizing the main conclusions, appropriate mesh resolutions are as follows: 0.002 m or finer for $p = 1$, 0.005 m or finer for $p = 2$, 0.01 m or finer for $p = 3$, 0.02 m or finer for $p = 4$ (that is all meshes considered.)

5.1.2. Resolution requirements for the onset time of instability

Next, the effect of resolution on the onset time of instabilities is evaluated. Before proceeding with the numerical results, a few comments on the computational meshes utilized are due. The computational grids used are refinements of the mesh depicted in Figure 8. This type of grids is coarser near the bottom of the domain, and allows for reduced computing times. Because instabilities arise in the top layer, the mesh is refined only there. Instabilities typically require long computing times to evolve, and this approach seemed to us effective without precluding accuracy. The base mesh used in computations is denoted X_0 and has elements in the diffusive layer of dimensions (in meters) of 0.05×0.02 , corresponding to the coarsest mesh resolution discussed in the previous section. We also compute on meshes denoted X_1 , X_2 , and X_3 , which correspond to refinements of X_0 obtained by subdividing each quadrilateral into 4, 16, and 64 quadrilaterals, respectively. Figure (9) and (10) show the same data from the results of the convergence study, emphasizing the effect of grid resolution (9) or increase in polynomial order (10). For completeness, Figure 9(d) shows the results obtained in [17] for different mesh resolutions (in meters). Considering Figure (9), we observe the following: For polynomial order $p = 2$ only the mesh X_3 yields reliable results; for the case $p = 3$, also the meshes X_2 and X_3 yields high-fidelity results; finally, in the case $p = 4$, also the meshes X_1 , X_2 , and X_3 deliver high-fidelity results. Virtually identical results were obtained imposing periodic boundary conditions instead of no-flow boundary conditions along the vertical boundaries.

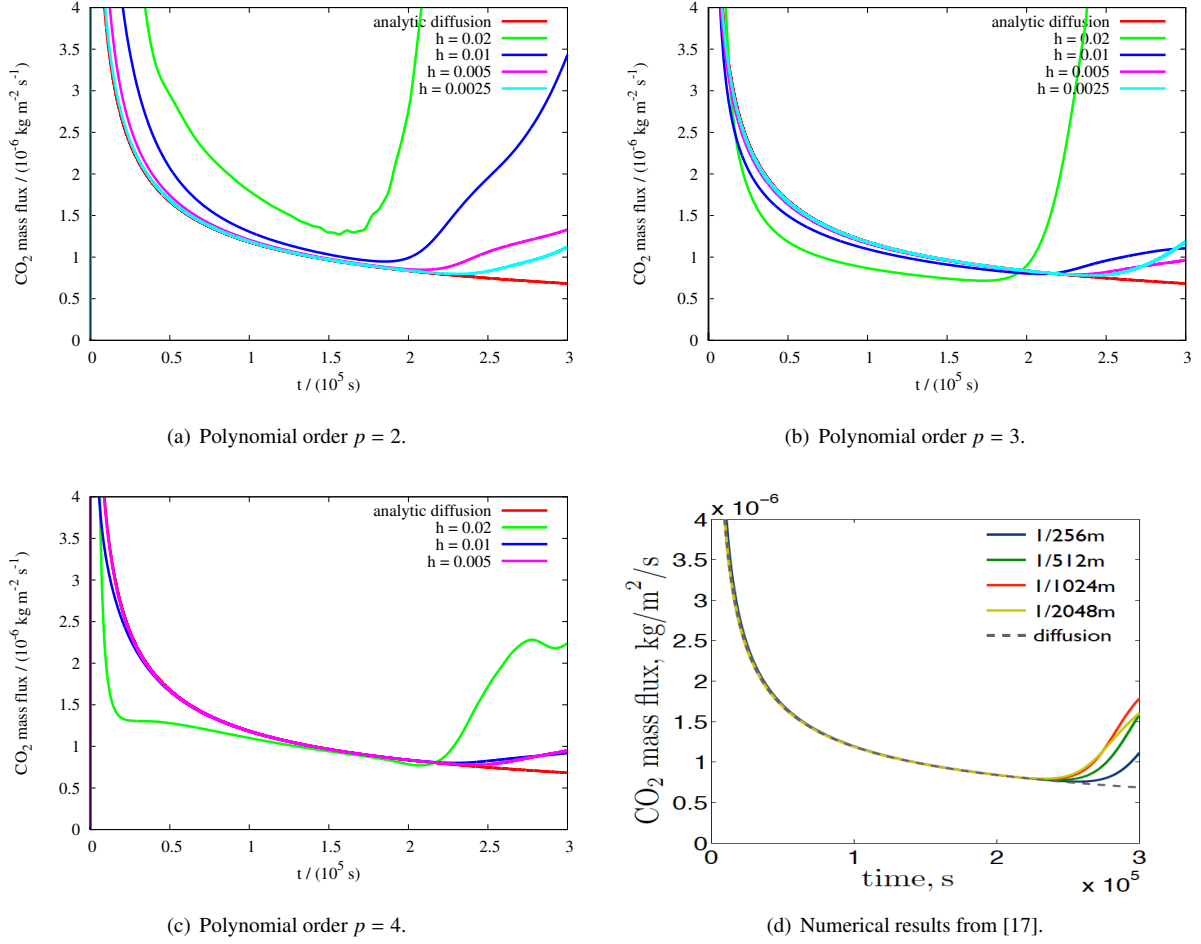


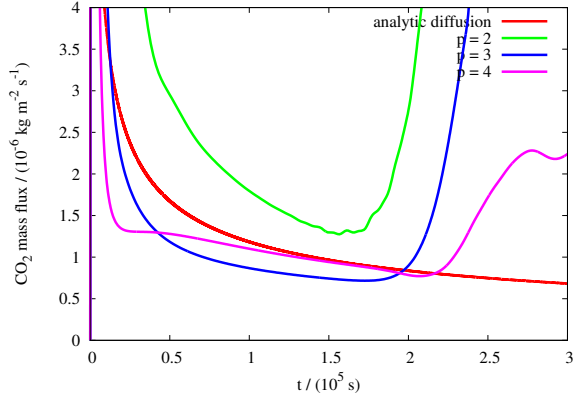
Figure 9: Time histories of the carbon dioxide flux at the top of the computational domain. One-dimensional study on the effect of polynomial order.

5.2. Long term behavior

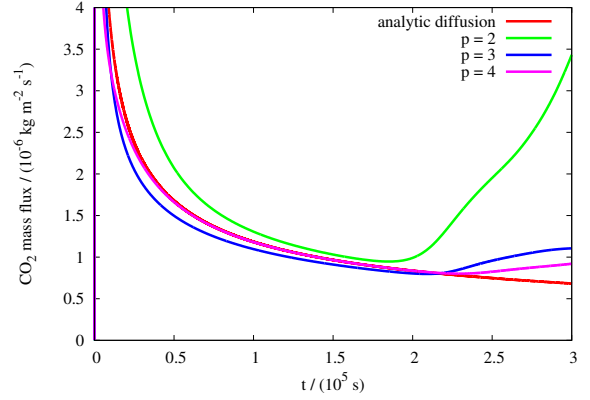
After their onset, viscous fingering instabilities develop into very complex nonlinear dynamics phenomena. A number of computations were performed to quantify the long-term behavior of the nonlinear system under consideration. The domain is the rectangle $\Omega = [0, 4] \times [-4, 0]$ (dimensions in meters), meshed with a base grid depicted in Figure 12, denoted $M0$, and also with a nested uniform refinement $M1$. Mesh $M0$ is composed of 160×50 elements, with a bias toward to top boundary, so that the vertical mesh length is 0.005 m, comparable with mesh $X2$ in the previous section, and clearly coarse toward the depth of the reservoir. Mesh $M1$ has been obtained by subdividing each of the quadrilaterals of mesh $M0$ into four quadrilaterals, and, as a result, the mesh length near the top boundary is the same of mesh $X3$. In practice, due to stretching, the resolutions of meshes $M0/M1$ are comparable with $X1/X2$, respectively, rather than $X2/X3$.

Remark11. The larger width of the computational domain Ω is necessary to accommodate a sufficient number of larger, fully developed finger plumes, as can also be seen in Figure 13. While this requirement is not necessary if the quantity of interest is the onset time of instability, it is more stringent in studies of long term behavior.

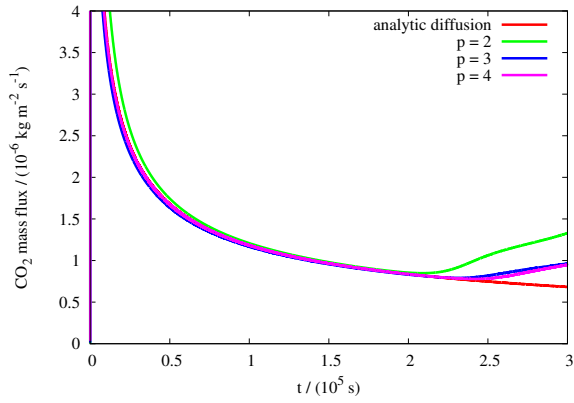
Figure 13 depicts snapshots of the carbon dioxide concentration at different times of the evolution of instabilities. The results were obtained using periodic boundary conditions along the vertical boundaries. Very similar results are obtained if, instead of periodic boundary conditions, homogenous Neumann (no-flow) boundary condition on the



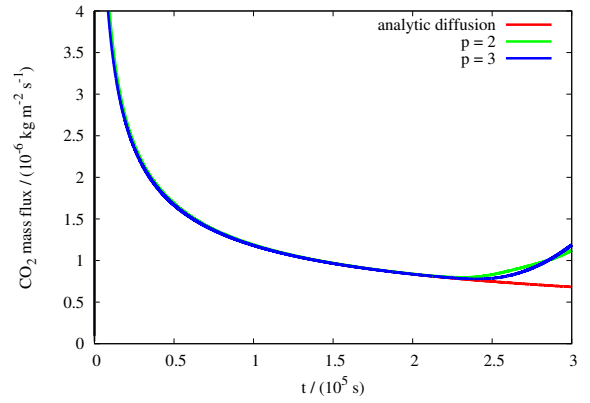
(a) Mesh size $h = 0.02$.



(b) Mesh size $h = 0.01$.



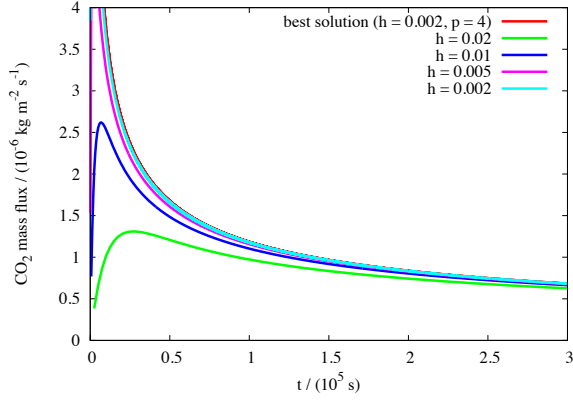
(c) Mesh size $h = 0.005$.



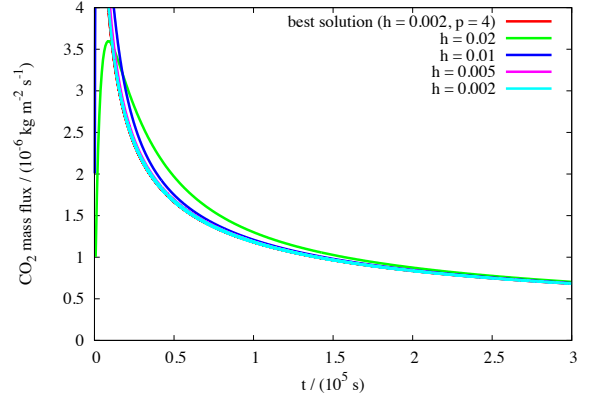
(d) Mesh size $h = 0.0025$.

Figure 10: Time histories of the carbon dioxide flux at the top of the computational domain. One-dimensional study on the effect of polynomial order.

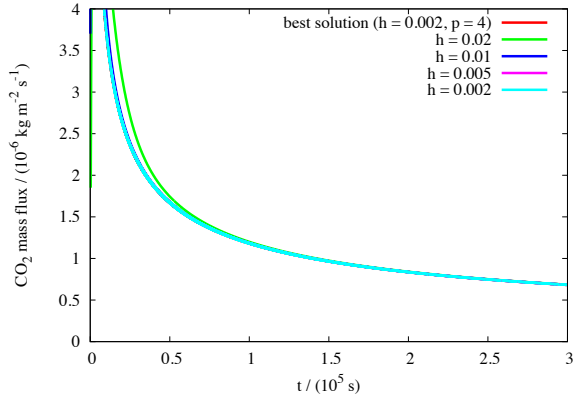
velocity \mathbf{u} are used. Figures 13(a) and 13(b) depict the early stages of the system, when the flow has not yet become unstable and only a sharp diffusive layer is present near the top boundary of the domain. In Figures 13(c) and 13(d), the onset of the instability is captured, corresponding to time $t = 2.5 \times 10^5 \text{ s} \approx 2.9$ days, as apparent also from Figures 9 and 10 in the previous section. It is worthwhile to compare with the time history of the carbon dioxide flux in Figure 14. Figure 14(a) shows in particular that computations performed with polynomial order $p = 2$ on the mesh X1 capture quite accurately the onset time, while in the case of $p = 4$ and X0, a slight delay is present, possibly due to the somewhat pronounced stretching of the mesh near the top boundary. It will be clear in what follows, however, that the computations with $p = 4$ seem more resolved in the center region of the computational domain, and are actually more accurate for the long term dynamics.



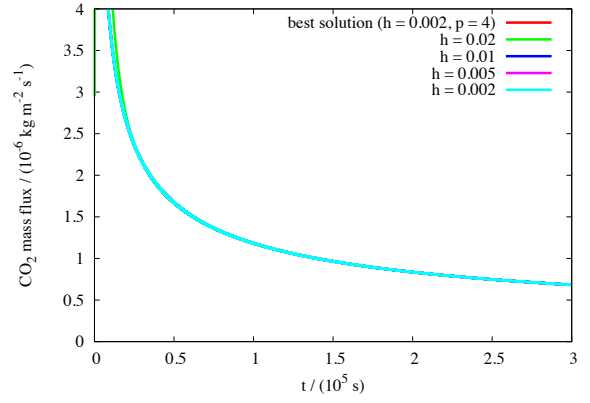
(a) Polynomial order $p = 1$.



(b) Polynomial order $p = 2$.



(c) Polynomial order $p = 3$.



(d) Polynomial order $p = 4$.

Figure 11: Time histories of the carbon dioxide flux at the top of the computational domain. One-dimensional study on the effect of resolution.

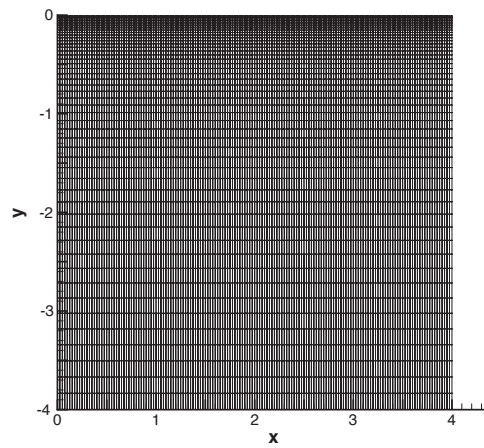


Figure 12: Sketch of the base mesh $M0$ used for the computation of onset times. The resolution near the upper boundary is comparable with the mesh $X2$ in the previous section. An additional refinement by splitting every element into four sub-elements was performed, and denoted mesh $M1$, with elements near the upper boundary of size comparable to $X3$.

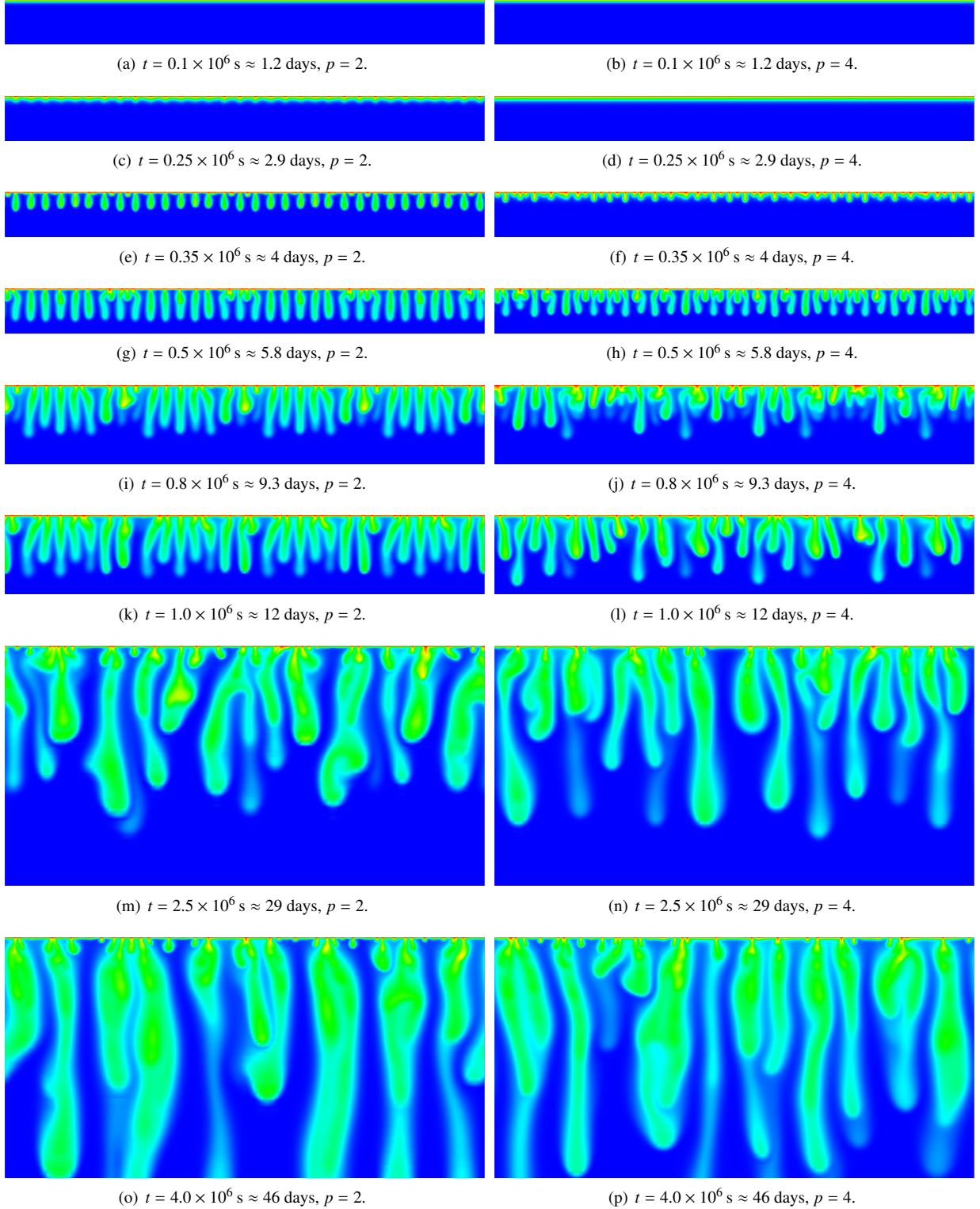


Figure 13: Snapshots of the concentration of carbon dioxide at different times for computations on mesh *M0* and $p = 2$ (left column) or $p = 4$ (right column). Homogenous boundary conditions (no flow) are used on the right and left vertical boundaries. The horizontal size is the same for all pictures (4 m), while the vertical size tracks the flow evolution (varying from approximately 0.5 m to 2 m). Computations were performed on a domain of $4\text{m} \times 4\text{m}$. Compare with Figure 14, where the time history of the carbon dioxide flux from the top boundary is computed.

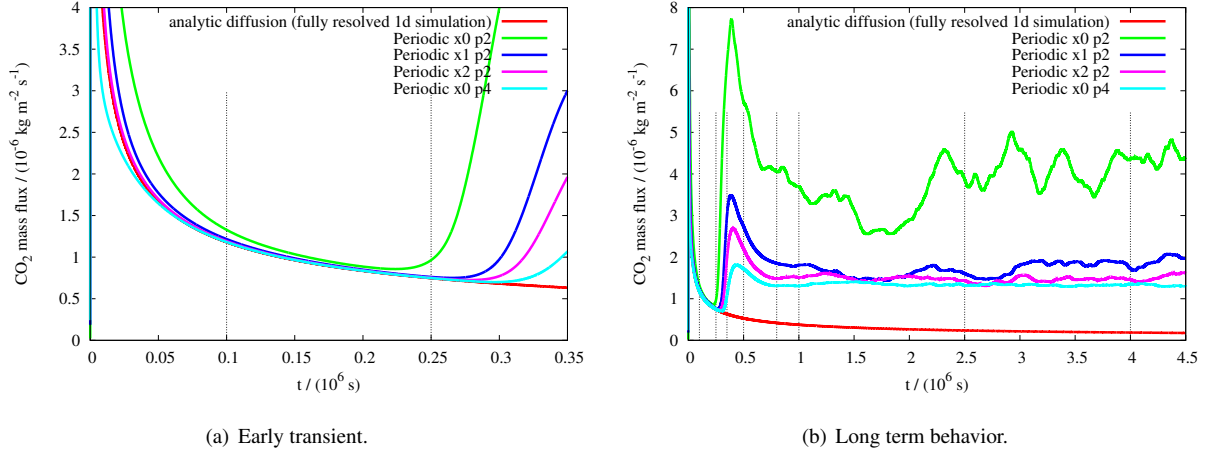


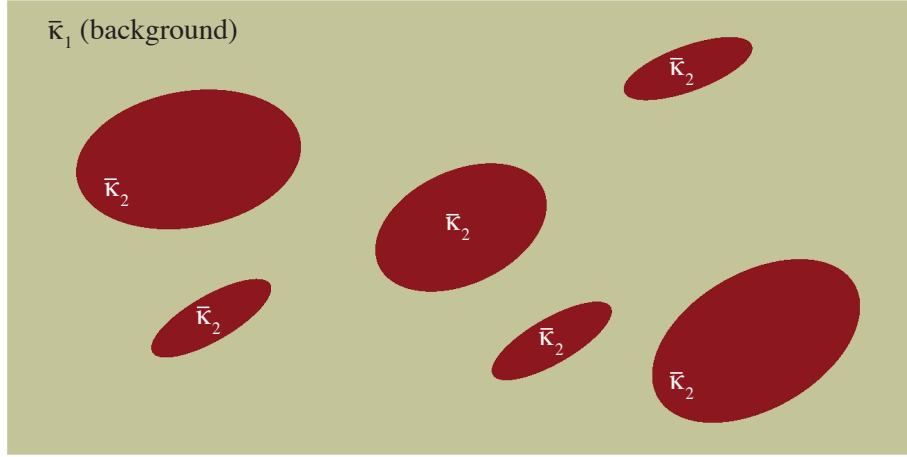
Figure 14: Time history of the carbon dioxide flux from the top boundary, for various resolutions and boundary conditions along the two vertical boundaries of the domain $\Omega = [0, 4] \times [-4, 0]$ (with dimensions in meters.)

Figures 13(e) and 13(f) are interesting since they capture the typical coalescence of smaller fingers into larger structures. This phenomenon is helped by the upwelling of lighter brine from the depths of the reservoir. In practice, preferential routes for the plumes of heavier brine form in the reservoir. The flow can sustain them for some time, or wash them away by upwelling of lighter brine. This dynamics is best understood considering the sequence of Figures 13(g) and 13(h), Figures 13(i) and 13(j), and Figures 13(k) and 13(l). Figures 13(m) and 13(n) show a later time, at which coalescence phenomena have created very large plumes, allowing “fresh” brine to surface in between. Figures 13(g) and 13(h) depict the conditions of the aquifer after about 46 days from the initiation of the dissolution process: The large structures have kept growing and are now attracting flow richer in carbon dioxide from above. At this stage the whole dynamics seems to have reached statistical stationarity, as can easily be appreciated observing the flux time history in Figure 14. In particular, it is visible the effect of different resolutions. Note that the average around which the flux seems to oscillate in the case $p = 4$ is about $1.5 \times 10^6 \text{ kg m}^{-2} \text{ s}^{-1}$, which is the same value found in the results in [17]. The computations with $p = 4$ also closely match the peak value of the flux after the instabilities occur, namely $2.0 \times 10^6 \text{ kg m}^{-2} \text{ s}^{-1}$.

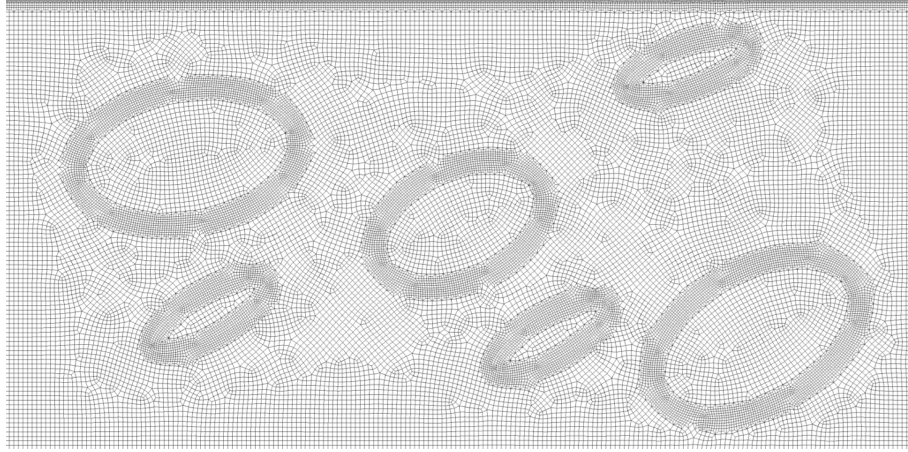
These results confirm that the presented computations with $p = 4$, although slightly improvable, can capture the salient dynamics of the long term behavior of viscous fingering instabilities for carbon dioxide applications. It is also worthwhile to observe, in Figure 14, the effect of resolution on the carbon dioxide flux: Under-resolved computations tend to over-predict the amount of CO_2 sequestered, both in terms of peak and average values of the flux of carbon dioxide sequestered. This final test case concludes the extensive verification of the proposed method against results from the available literature.

6. Unstructured computational capabilities

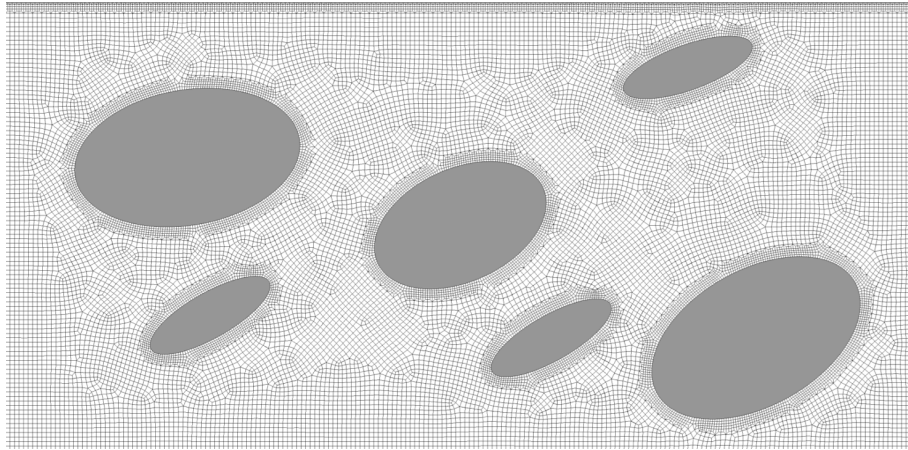
One of the key features of the proposed approach with respect to existing methods for viscous fingering computations in porous media is the ability to accommodate for the often complex subsurface geometry without resorting to embedded/immersed boundary techniques. The latter methodologies are viable, but tend to lower the accuracy of the solution near the immersed/embedded boundaries, which in the case of viscous fingering instabilities are an integral part of the system dynamics. We present next tests to show the new possibilities that discontinuous Galerkin methods offer. Consider the domain depicted in Figure 15(a): A number of elliptical obstructions are embedded in a domain Ω of size $4\text{m} \times 2\text{m}$. The average permeability of the background domain is $\bar{\kappa}_1 = 10\text{D} = 9.869233 \times 10^{-12} \text{ m}^2$, and we have considered three possible values for the average permeability of the obstructions: $\bar{\kappa}_2 = \bar{\kappa}_1$ (no obstructions, just an unstructured mesh), $\bar{\kappa}_2 = \bar{\kappa}_1/10$ (obstructions with moderate permeability), and $\bar{\kappa}_2 = 0$ (fully impermeable obstructions.)



(a) Sketch of the complex geometry domain and distribution of permeability in the general case.



(b) Mesh detail in the case $\bar{\kappa}_2 = \bar{\kappa}_1/10$.



(c) Mesh detail in the case $\bar{\kappa}_2 = 0$, for which the domain has impermeable obstructions.

Figure 15: Geometry and meshes for a general domain with embedded elliptical obstructions. The subsurface model is constituted of a rectangular domain of size $4\text{m} \times 2\text{m}$, in which are positioned a number of elliptical obstructions of different permeabilities. Details of the meshes in the cases $\bar{\kappa}_2 = \bar{\kappa}_1/10$ (less permeable obstructions) and $\bar{\kappa}_2 = 0$ (impermeable obstructions) are also shown.

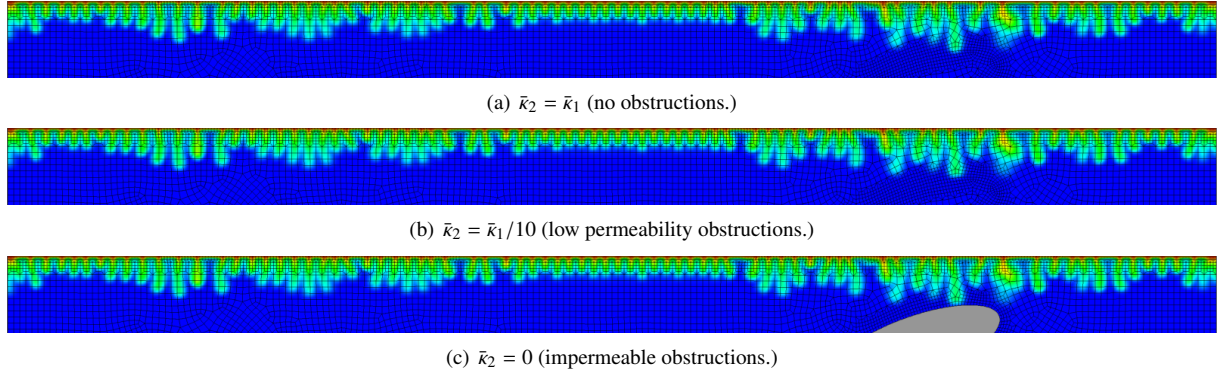


Figure 16: Snapshots of the concentration of carbon at time $t = 0.3 \times 10^6 \text{ s} \approx 3.5$ days, for the three distribution of permeability \bar{k}_2 . Compare with Figure 17, where the time history of the carbon dioxide flux from the top boundary is computed.

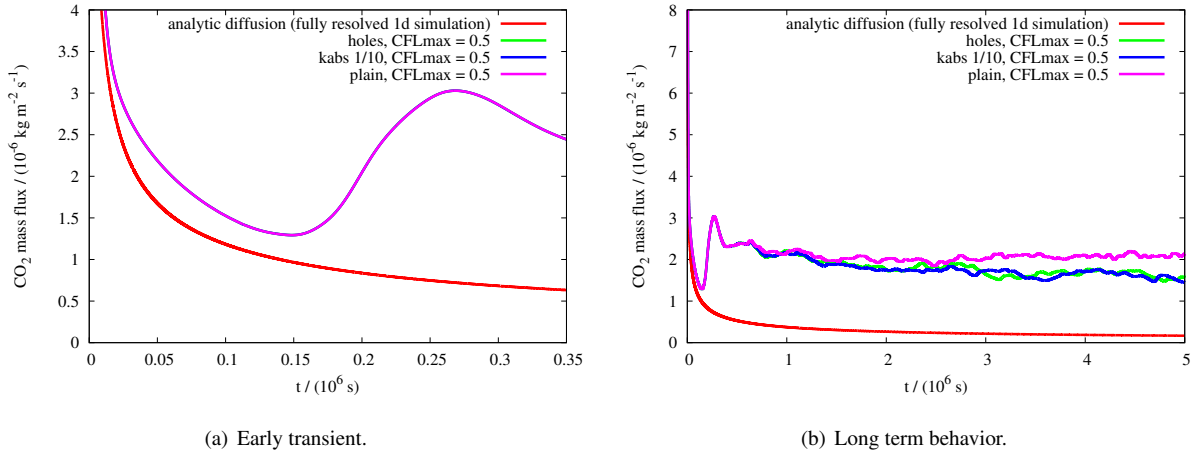
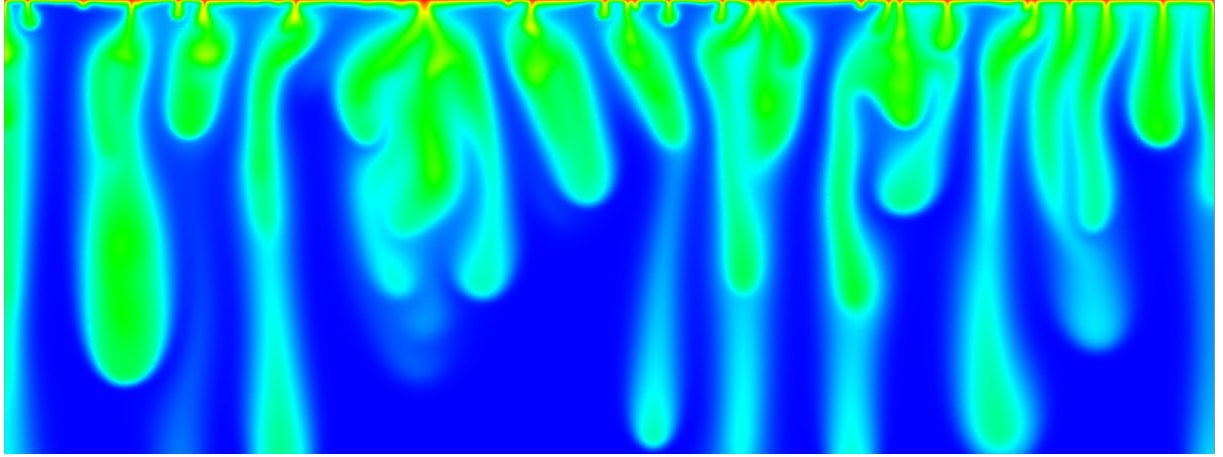


Figure 17: Time history of the carbon dioxide flux from the top boundary, for the computations presented in Figures 16, 18, and 19.)

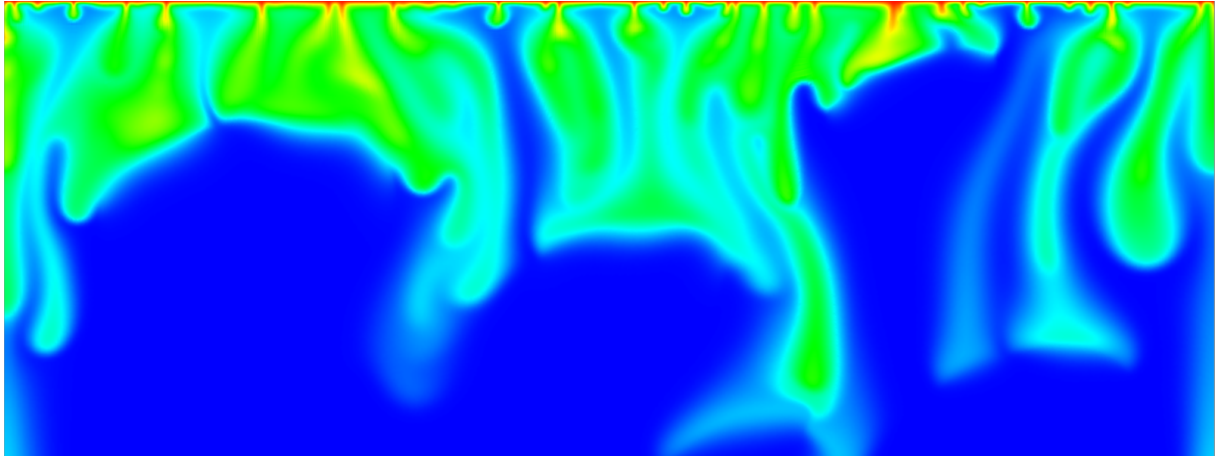
All other parameters, including the permeability perturbation are identical to the specification of the instability problem described in Section 5. All computations were performed using quadratic polynomials on fully unstructured meshes.

Figure 16 shows an early stage of the computations: The flow has just reached the obstructions, and very little difference is observed in the flow patterns for the three different values of the permeability \bar{k}_2 . Interestingly, also the values of the average flux from the top of the domain is virtually identical, as seen in Figure 17(a). The same picture indicates that the resolution near the upper boundary is not sufficient to fully capture viscous boundary layer, and that this could also be the case in the rest of the domain (the long-term average in the case of no obstructions is a little higher than the value 1.5, see Figure 17(b).) Note however that the purpose of these tests is not to fully resolve the phenomena under investigation, but instead to highlight the new capabilities that the discontinuous Galerkin approaches discloses.

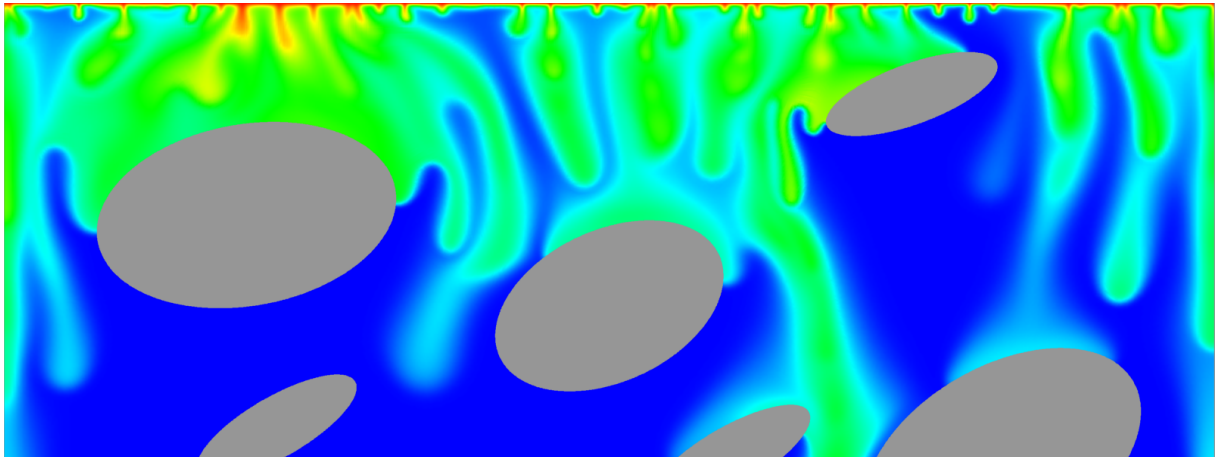
Later on in the computations, the obstructions deflect the flow in preferential channels, and the overall flux of carbon dioxide from the top boundary is reduced, although not very significantly (about 20%.) It is also interesting to note that there is very little difference in the carbon dioxide average flux between the cases of $\bar{k}_2 = \bar{k}_1/10$ and $\bar{k}_2 = 0$ (impermeable obstructions.)



(a) $\bar{\kappa}_2 = \bar{\kappa}_1$ (no obstructions.)

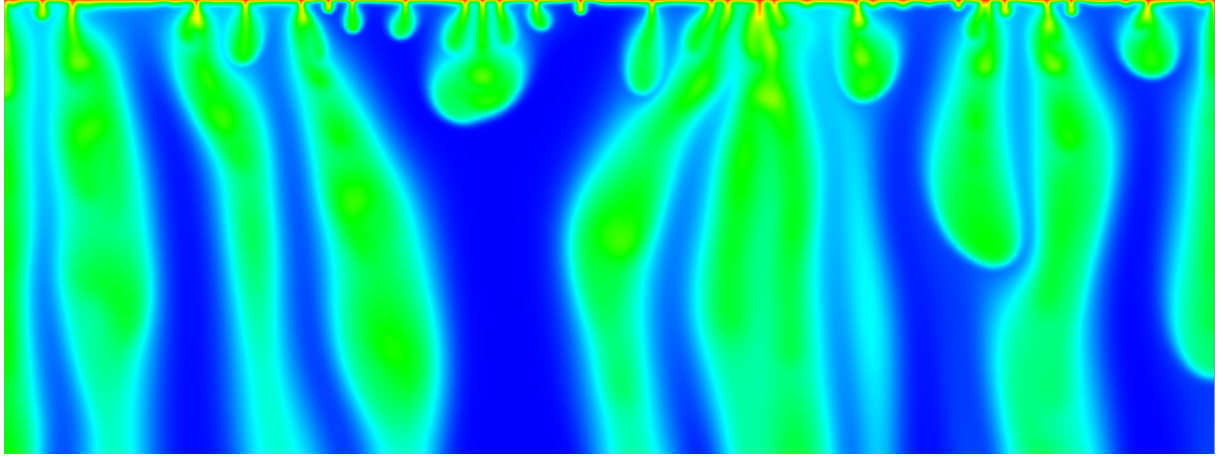


(b) $\bar{\kappa}_2 = \bar{\kappa}_1 / 10$ (low permeability obstructions.)

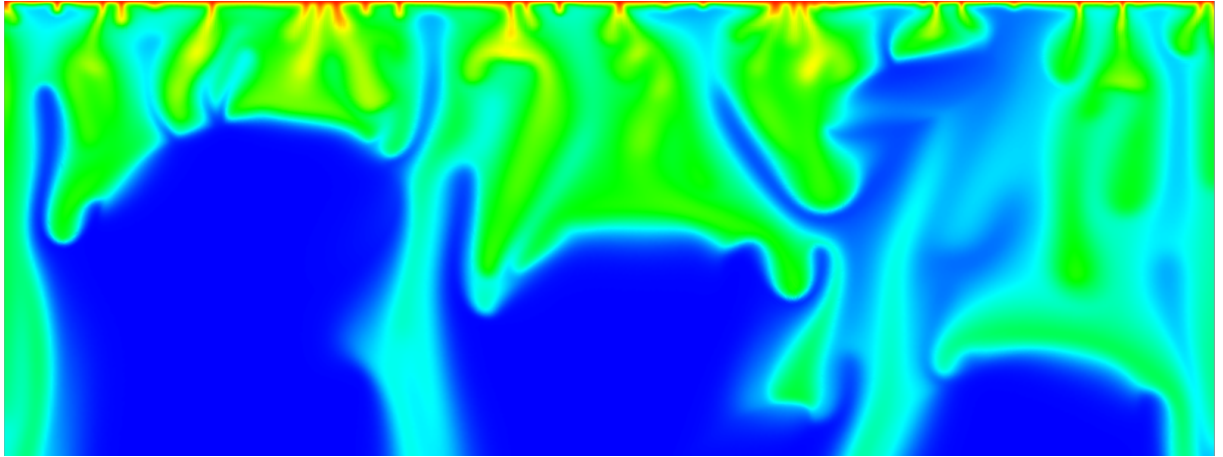


(c) $\bar{\kappa}_2 = 0$ (impermeable obstructions.)

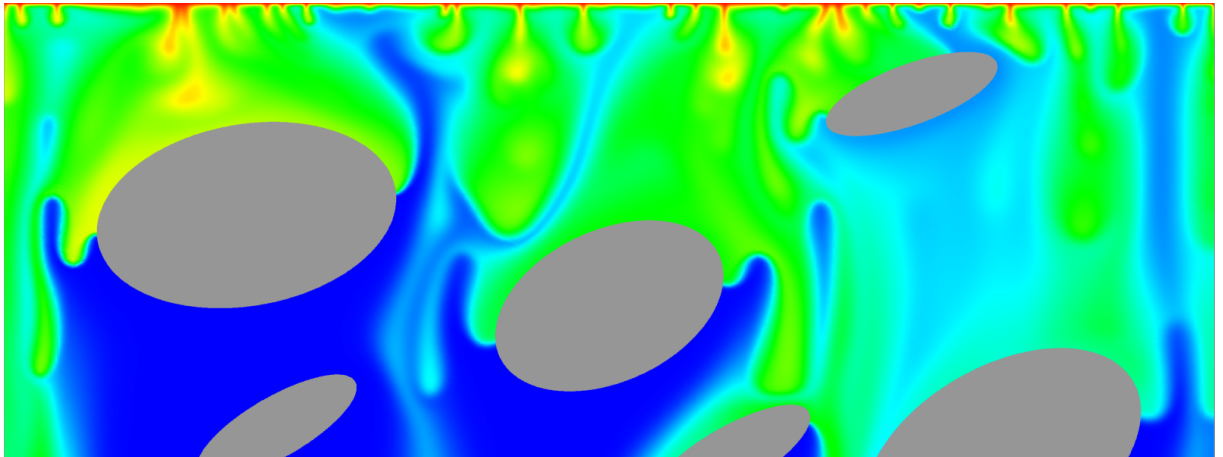
Figure 18: Snapshots of the concentration of carbon at time $t = 2.6 \times 10^6 \text{ s} \approx 30$ days, for the three distribution of permeability $\bar{\kappa}_2$. Compare with Figure 17, where the time history of the carbon dioxide flux from the top boundary is computed.



(a) $\bar{\kappa}_2 = \bar{\kappa}_1$ (no obstructions.)

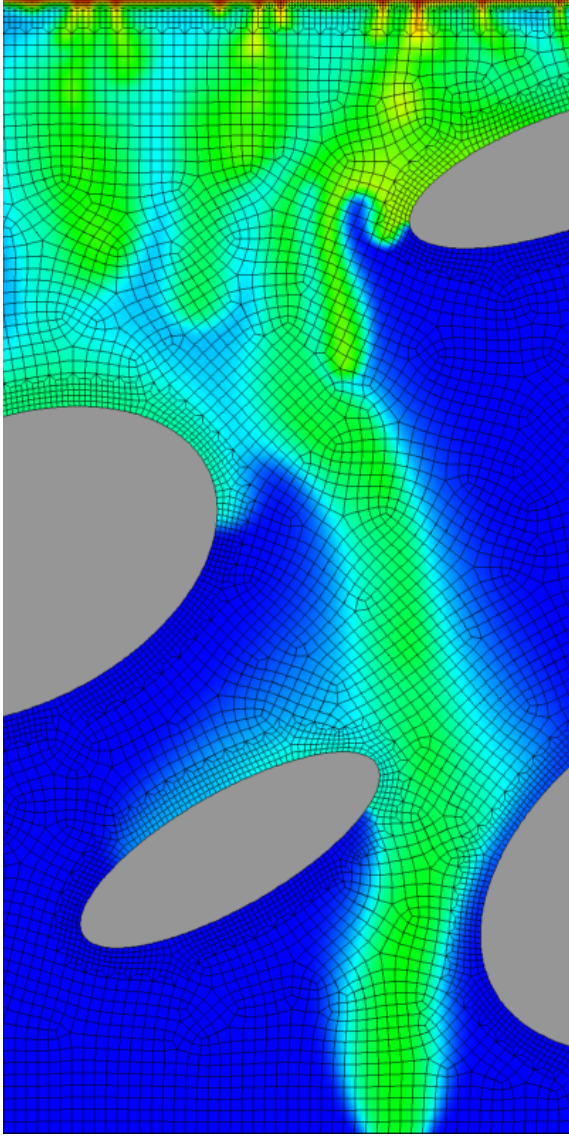


(b) $\bar{\kappa}_2 = \bar{\kappa}_1/10$ (low permeability obstructions.)

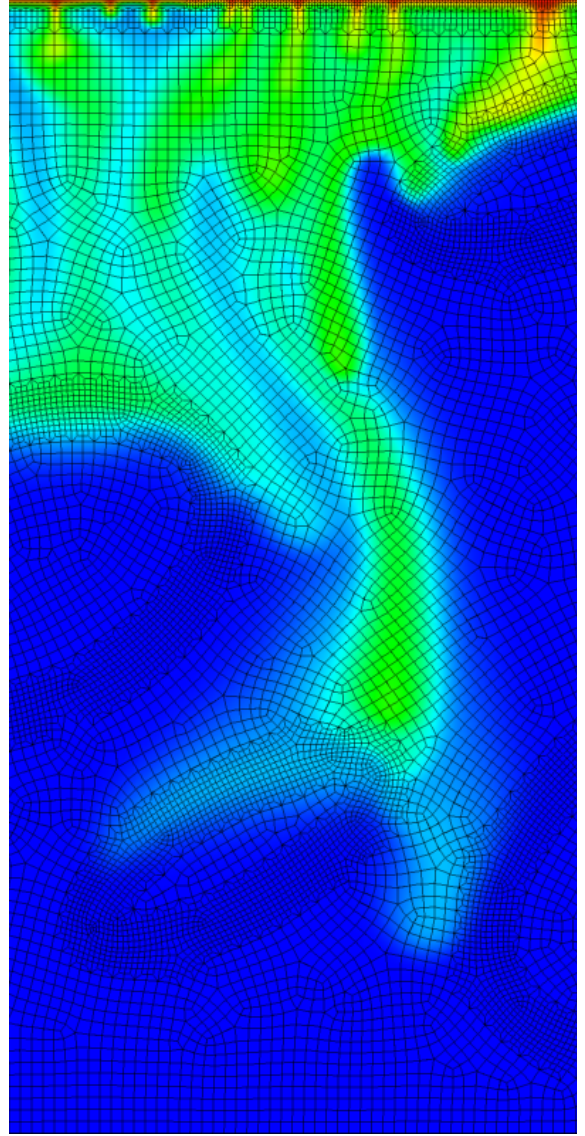


(c) $\bar{\kappa}_2 = 0$ (impermeable obstructions.)

Figure 19: Snapshots of the concentration of carbon at time $t = 5.0 \times 10^6$ s \approx 58 days, for the three distribution of permeability $\bar{\kappa}_2$. Compare with Figure 17, where the time history of the carbon dioxide flux from the top boundary is computed.



(a) $\bar{\kappa}_2 = 0$ (impermeable obstructions.)



(b) $\bar{\kappa}_2 = \bar{\kappa}_1/10$ (low permeability obstructions.)

Figure 20: A detail at $t = 2.6 \times 10^6 \text{ s} \approx 30$ days of the domain with the concentration distribution and the computational mesh. It is clearly noticeable the lack of oscillations and the complexity of the computational grid.

While the chosen shapes of the obstruction are somewhat arbitrary, their smooth contours allow for very interesting phenomena to occur, such as the oscillating shedding of viscous plumes from the boundary of the obstructions. This results in complex umbrella-like structures which develop from the competition between inertia and buoyancy forces. This can be seen more clearly in Figures 18 and 19. On the same figures and Figure 20, it is also easy to appreciate the smoothness of the solution. This is a quite interesting result, especially in the case of Figure 20(b), considering that the element size of the unstructured mesh changes quite rapidly near the regions where an abrupt variation in permeability occurs.

7. Summary

In this work, a new high-order discontinuous Galerkin method for nonlinear computations of gravity driven instabilities has been presented. The key feature of the method is the ability to deliver high-order solutions on fully unstructured meshes. We demonstrated with extensive verification tests these claims. We consider this capability of great importance, especially when considering carbon sequestration in brine aquifers for realistic subsurface geometries. A number of additional publications are in preparation on our ongoing efforts to extensively test the computational capabilities on unstructured grids and geometrically complex computational domains, as well as algebraic multigrid strategies to accelerate the solution of the Darcy flow system. We hope to report about these advances in the very near future.

Acknowledgments

The authors would like to acknowledge the support of Department of Energy’s Office of Science through the SciDAC-e Research Grant “Algebraic Multi-Grid Methods for Modeling and Simulation of Carbon Sequestration Processes on Multi-Core/GPU Architectures,” No. 10-014677.

References

- [1] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM Journal on Numerical Analysis*, 39:1749–1779, 2002.
- [2] F. Brezzi, L. D. Marini, and E. Süli. Discontinuous Galerkin methods for first-order hyperbolic problems. *Mathematical Models and Methods in Applied Sciences*, 14:1893–1903, 2004.
- [3] B. Cockburn and C.-W. Shu. The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35:2440–2463, 1998.
- [4] J. P. Ennis-King, I. Preston, and L. Paterson. Onset of convection in anisotropic porous media subject to a rapid change in boundary conditions. *SPE Journal*, 17:349–356, 2005.
- [5] V. Girault, S. Sun, M. F. Wheeler, and I. Yotov. Coupling discontinuous Galerkin and mixed finite element discretizations using mortar finite elements. *SIAM Journal on Numerical Analysis*, 46:949–979, 2008.
- [6] J. Gounot and J. P. Caltagirone. Stabilité et convection naturelle au sein d’une couche poreuse non homogène. *International Journal of Heat and Mass Transfer*, 32:1131–1140, 1989.
- [7] J. J. Hidalgo and J. Carrera. Effect of dispersion on the onset of convection during CO₂ sequestration. *Journal of Fluid Mechanics*, 640:441–452, 2009.
- [8] J. J. Hidalgo, J. Carrera, and A. Medina. Role of salt sources in density-dependent flow. *Water Resources Research*, 45:W05503, 2009.
- [9] H. Hoteit and A. Firoozabadi. Multicomponent fluid flow by discontinuous Galerkin and mixed methods in unfractured and fractured media. *Advances in Water Resources*, 41:W11412, 2005.
- [10] Jr. J. Douglas and T. Dupont. Interior penalty procedures for elliptic and parabolic Galerkin methods. In *Lecture Notes in Physics*, volume 58. Springer-Verlag, Berlin, 1976.
- [11] K. Johannsen. On the validity of the Boussinesq approximation for the Elder problem. *Computational Geosciences*, 7:169–182, 2003.
- [12] J. Douglas Jr., D. Frias, N. Henderson, and F. Pereira. Simulation of single-phase multicomponent flow problems in gas reservoirs by Eulerian-Lagrangian techniques. *Transport in Porous Media*, 50:307–342, 2002.
- [13] A. Landman and R. Schotting. Heat and brine transport in porous media: the Oberbeck-Boussinesq approximation revisited. *Transport in Porous Media*, 70:355–373, 2007.
- [14] R. Liu, M.F. Wheeler, C.N. Dawson, and R.H. Dean. Modeling of convection-dominated thermoporomechanics problems using incomplete interior penalty Galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 198:912–919, 2009.
- [15] R. Liu, M.F. Wheeler, C.N. Dawson, and R.H. Dean. On a coupled discontinuous/continuous Galerkin framework and an adaptive penalty scheme for poroelasticity problems. *Computer Methods in Applied Mechanics and Engineering*, 198:3499–3510, 2009.
- [16] G. S. H. Pau, A. S. Almgren, J. B. Bell, and M. J. Lijewski. A parallel second-order adaptive mesh algorithm for incompressible flow in porous media. *Phil. Trans. R. Soc. A*, 367:4633–4654, 2009.
- [17] G. S. H. Pau, J. B. Bell, K. Pruess, A. S. Almgren, M. J. Lijewski, K. Zhang, C. Yang, and Y. Gu. High-resolution simulation and characterization of density-driven flow in CO₂ storage in saline aquifers. *Advances in Water Resources*, 33:443–455, 2010.
- [18] P. J. Phillips and M. F. Wheeler. A coupling of mixed and discontinuous Galerkin finite-element methods for poroelasticity. *Computational Geosciences*, 12:417–435, 2008.
- [19] A. Riaz, M. Hesse, H. A. Tchelepi, and F. M. Orr. Onset of convection in a gravitationally unstable diffusive boundary layer in porous media. *Journal of Fluid Mechanics*, 548:87–111, 2006.
- [20] G. R. Richter. The discontinuous Galerkin method with diffusion. *Mathematics of Computation*, 58:631–643, 1992.
- [21] B. Rivière and M. F. Wheeler. Discontinuous Galerkin methods for flow and transport problems in porous media. *Communications in Numerical Methods in Engineering*, 18:63–68, 2002.
- [22] B. Rivière, M. F. Wheeler, and K. Banas. Part II. Discontinuous Galerkin method applied to a single phase flow in porous media. *Computational Geosciences*, 4:337–349, 2000.

- [23] B. Rivière, M. F. Wheeler, and V. Girault. Elliptic collocation-finite element method with interior penalties. *SIAM Journal on Numerical Analysis*, 15:152–161, 1978.
- [24] B. Rivière, M. F. Wheeler, and V. Girault. Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. Part I. *Computational Geosciences*, 3:337–360, 1999.
- [25] B. Rivière, M. F. Wheeler, and V. Girault. A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems. *SIAM Journal on Numerical Analysis*, 39:902–931, 2001.
- [26] B. Rivière, M. F. Wheeler, and V. Girault. A priori error estimates for finite element methods based on discontinuous approximation spaces for elliptic problems. *SIAM Journal on Numerical Analysis*, 39:902–931, 2001.
- [27] S. Sun and M. F. Wheeler. Anisotropic and dynamic mesh adaptation for discontinuous Galerkin methods applied to reactive transport. *Computer Methods in Applied Mechanics and Engineering*, 195:3382–3405, 2006.
- [28] S. Sun and M. F. Wheeler. Local problem-based a posteriori error estimators for discontinuous Galerkin approximations of reactive transport. *Computers and Mathematics with Applications*, 52:637–650, 2006.
- [29] S. Sun and M. F. Wheeler. Discontinuous Galerkin methods for simulating bioreactive transport of viruses in porous media. *Advances in Water Resources*, 30:1696–1710, 2007.
- [30] S. Sun and M. F. Wheeler. Local problem-based a posteriori error estimators for discontinuous Galerkin approximations of reactive transport. *Computational Geosciences*, 11:87–101, 2007.
- [31] S. Y. Sun and M. F. Wheeler. Symmetric and nonsymmetric discontinuous galerkin methods for reactive transport in porous media. *SIAM Journal on Numerical Analysis*, 43:195–219, 2005.
- [32] S. Y. Sun and M. F. Wheeler. A posteriori error estimation and dynamic adaptivity for symmetric discontinuous Galerkin approximations of reactive transport problems. *Computer Methods in Applied Mechanics and Engineering*, 195:632–652, 2006.
- [33] X. Xu, S. Chen, and D. Zhang. Convective stability analysis of the long-term storage of carbon dioxide in deep saline aquifers. *Advances in Water Resources*, 29:397–407, 2006.